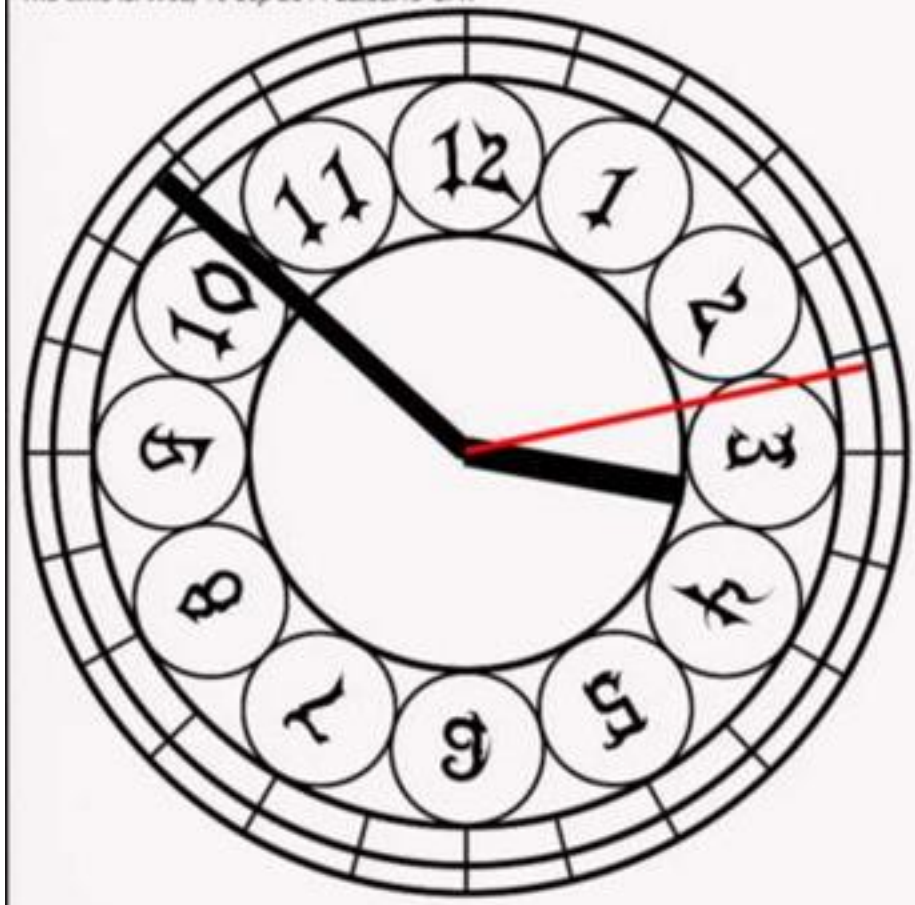


Old Style Clock App

The time is: Wed, 10 Sep 2014 22:52:13 GMT



default.htm

```
1 <!DOCTYPE html>
2
3 <html lang="en">
4 <head>
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta charset="utf-8" />
7   <title>Old Style Clock App</title>
8   <link rel="stylesheet" href="app.css" type="text/css" />
9   <script src="clock.js"></script>
10  <script src="app.js"></script>
11 </head>
12 <body>
13   <h1>Old Style Clock App</h1>
14
15   <div id="content"></div>
16 </body>
17 </html>
```

Watches

Add watch

Call stack

Breakpoints

```
1  /// 
2
3  module Clock {
4      export class App {
5          element: HTMLElement;
6          span: HTMLElement;
7          timerToken: number;
8          clock: Clock.ClockFace;
9
10         constructor(element: HTMLElement) {
11             this.element = element;
12             this.element.innerHTML += "The time is: ";
13             this.span = document.createElement('span');
14             this.element.appendChild(this.span);
15             this.span.innerText = new Date().toUTCString();
16
17             var clockDiv = document.createElement("div");
18             element.appendChild(clockDiv);
19             this.clock = new ClockFace(clockDiv);
20         }
21
22         start() {
23             this.timerToken = setInterval(() => this.span.innerHTML = new Date().toUTCString(), 1000);
24         }
25
26         stop() {
27             clearTimeout(this.timerToken);
28         }
29     }
30 }
31
32
33 window.onload = () => {
34     var el = document.getElementById('content');
35     var greeter = new Clock.App(el);
36     greeter.start();
37 }
```

this	[object App]
arguments	[object (Arguments)]
element	[object HTMLDivElement]
clockDiv	[object HTMLDivElement]
[Scope]	[Anonymous function]
[Globals]	
Add watch	

App	app.ts (19, 13)
window.onload	app.ts (35, 5)

```
1  //<script src="path"/>Clock.js />
2  var Clock;
3  (function (Clock) {
4      var App = (function () {
5          function App(element) {
6              this.element = element;
7              this.element.innerHTML += "The time is: ";
8              this.span = document.createElement('span');
9              this.element.appendChild(this.span);
10             this.span.innerHTML = new Date().toUTCString();
11
12             var clockDiv = document.createElement("div");
13             element.appendChild(clockDiv);
14             this.clock = new Clock.ClockFace(clockDiv);
15         }
16         App.prototype.start = function () {
17             var _this = this;
18             this.timerToken = setInterval(function () {
19                 return _this.span.innerHTML = new Date().toUTCString();
20             }, 1000);
21         };
22
23         App.prototype.stop = function () {
24             clearInterval(this.timerToken);
25         };
26         return App;
27     })();
28     Clock.App = App;
29 })(Clock || (Clock = {}));
30
31 window.onload = function () {
32     var el = document.getElementById('content');
33     var greeter = new Clock.App(el);
34     greeter.start();
35 };
36 //<script src="path"/>app.js.map
37
```

this	[object App]
arguments	[object Arguments]
element	[object HTMLDivElement]
clockDiv	[object HTMLDivElement]
[Scope]	[Anonymous function]
[Globals]	
Add watch	

App	app.js (14, 12)
window.onload	app.js (33, 5)

default.htm Clocks app.js

```
1 module Clock {
2   export class ClockFace {
3     private static Uri: string = "../img/clock1.png";
4     private static MinuteIncrement: number = 6;
5     private static HourIncrement: number = 30;
6
7     private element: HTMLDivElement;
8     private hourHand: HTMLElement;
9     private minuteHand: HTMLElement;
10    private secondHand: HTMLElement;
11
12    constructor(element: HTMLDivElement) {
13      var image: HTMLImageElement = document.createElement("img");
14      image.src = ClockFace.Uri;
15
16      element.classList.add("clockFace");
17      element.appendChild(image);
18
19      // Create hands
20      this.hourHand = document.createElement("div");
21      this.hourHand.classList.add("hourHand");
22      element.appendChild(this.hourHand);
23
24      this.minuteHand = document.createElement("div");
25      this.minuteHand.classList.add("minuteHand");
26      element.appendChild(this.minuteHand);
27
28      this.secondHand = document.createElement("div");
29      this.secondHand.classList.add("secondHand");
30      element.appendChild(this.secondHand);
31
32      setInterval(() => {
33        var date = new Date();
34
35        var hours = date.getHours() % 12;
36        var minutes = date.getMinutes();
37        var seconds = date.getSeconds();
38
39        this.hourHand.style.transform = "rotate(" + (hours * ClockFace.HourIncrement + (minutes * ClockFace.MinuteIncrement / 30)) + "deg)";
40        this.minuteHand.style.transform = "rotate(" + minutes * ClockFace.MinuteIncrement + "deg)";
41        this.secondHand.style.transform = "rotate(" + seconds * ClockFace.MinuteIncrement + "deg)";
42      }, 1000);
43    }
44  }
45 }
46 }
47 }
```

Watches

[Locals]

this	[object (App)]
arguments	[object (Arguments)]
element	[object HTMLDivElement]
clockDiv	[object HTMLDivElement]
[Scope]	[Anonymous function]
[Global]	
Add watch	

Call stack

Breakpoints

Main frame

App	app.js (14, 13)
window.onload	app.js (33, 5)

```
1  //<script src="Clock.js">/>
2  var Clock;
3  (function (Clock) {
4      var App = (function () {
5          function App(element) {
6              this.element = element;
7              this.element.innerHTML += "The time is: ";
8              this.span = document.createElement('span');
9              this.element.appendChild(this.span);
10             this.span.innerHTML = new Date().toUTCString();
11
12             var clockDiv = document.createElement("div");
13             element.appendChild(clockDiv);
14             this.clock = new Clock.ClockFace(clockDiv);
15         }
16         App.prototype.start = function () {
17             var _this = this;
18             this.timerToken = setInterval(function () {
19                 return _this.span.innerHTML = new Date().toUTCString();
20             }, 1000);
21         };
22
23         App.prototype.stop = function () {
24             clearInterval(this.timerToken);
25         };
26         return App;
27     })();
28     Clock.App = App;
29 })(Clock || (Clock = {}));
30
31 window.onload = function () {
32     var el = document.getElementById('content');
33     var greeter = new Clock.App(el);
34     greeter.start();
35 };
36 //<script src="app.js">/>
37
```

[Locals]

this	[object App]
arguments	[object Arguments]
element	[object HTMLDivElement]
clockDiv	[object HTMLDivElement]
[Scope]	[Anonymous function]
[Global]	
Add watch	

Call stack

Breakpoints

Main frame

App	app.js (14, 12)
window.onload	app.js (33, 5)

default.htm Clocks app.js

```
1  /// sourceMappingURL=app.js.map
2  var Clock;
3  (function (Clock) {
4      var App = (function () {
5          function App(element) {
6              this.element = element;
7              this.element.innerHTML += "The time is: ";
8              this.span = document.createElement('span');
9              this.element.appendChild(this.span);
10             this.span.innerHTML = new Date().toUTCString();
11
12             var clockDiv = document.createElement('div');
13             element.appendChild(clockDiv);
14             this.clock = new Clock.ClockFace(clockDiv);
15         }
16         App.prototype.start = function () {
17             var _this = this;
18             this.timerToken = setInterval(function () {
19                 return _this.span.innerHTML = new Date().toUTCString();
20             }, 1000);
21         };
22
23         App.prototype.stop = function () {
24             clearInterval(this.timerToken);
25         };
26         return App;
27     })();
28     Clock.App = App;
29 })(Clock || (Clock = {}));
30
31 window.onload = function () {
32     var el = document.getElementById('content');
33     var greeter = new Clock.App(el);
34     greeter.start();
35 };
36 //# sourceMappingURL=app.js.map
37
```

Watches

[Locals]

this	[object (App)]
arguments	[object (Arguments)]
element	[object HTMLDivElement]
clockDiv	[object HTMLDivElement]
[Scope]	[Anonymous function]
[Globals]	
Add watch	

Call stack

Breakpoints

Main frame

App	app.js (14, 13)
window.onload	app.js (33, 5)

default.htm Clocks app.js

```
1 module Clock {
2   export class ClockFace {
3     private static Uri: string = "../img/clock1.png";
4     private static MinuteIncrement: number = 6;
5     private static HourIncrement: number = 30;
6
7     private element: HTMLDivElement;
8     private hourHand: HTMLElement;
9     private minuteHand: HTMLElement;
10    private secondHand: HTMLElement;
11
12    constructor(element: HTMLDivElement) {
13      var image: HTMLImageElement = document.createElement("img");
14      image.src = ClockFace.Uri;
15
16      element.classList.add("clockFace");
17      element.appendChild(image);
18
19      // Create hands
20      this.hourHand = document.createElement("div");
21      this.hourHand.classList.add("hourHand");
22      element.appendChild(this.hourHand);
23
24      this.minuteHand = document.createElement("div");
25      this.minuteHand.classList.add("minuteHand");
26      element.appendChild(this.minuteHand);
27
28      this.secondHand = document.createElement("div");
29      this.secondHand.classList.add("secondHand");
30      element.appendChild(this.secondHand);
31
32      setInterval(() => {
33        var date = new Date();
34
35        var hours = date.getHours() % 12;
36        var minutes = date.getMinutes();
37        var seconds = date.getSeconds();
38
39        this.hourHand.style.transform = `rotate(${(hours * ClockFace.HourIncrement / 30) * "deg"});
40        this.minuteHand.style.transform = `rotate(${(minutes * ClockFace.MinuteIncrement * "deg"});
41        this.secondHand.style.transform = `rotate(${(seconds * ClockFace.MinuteIncrement * "deg"});
42      }, 1000);
43    }
44  }
45 }
46 }
47 }
```

Watches

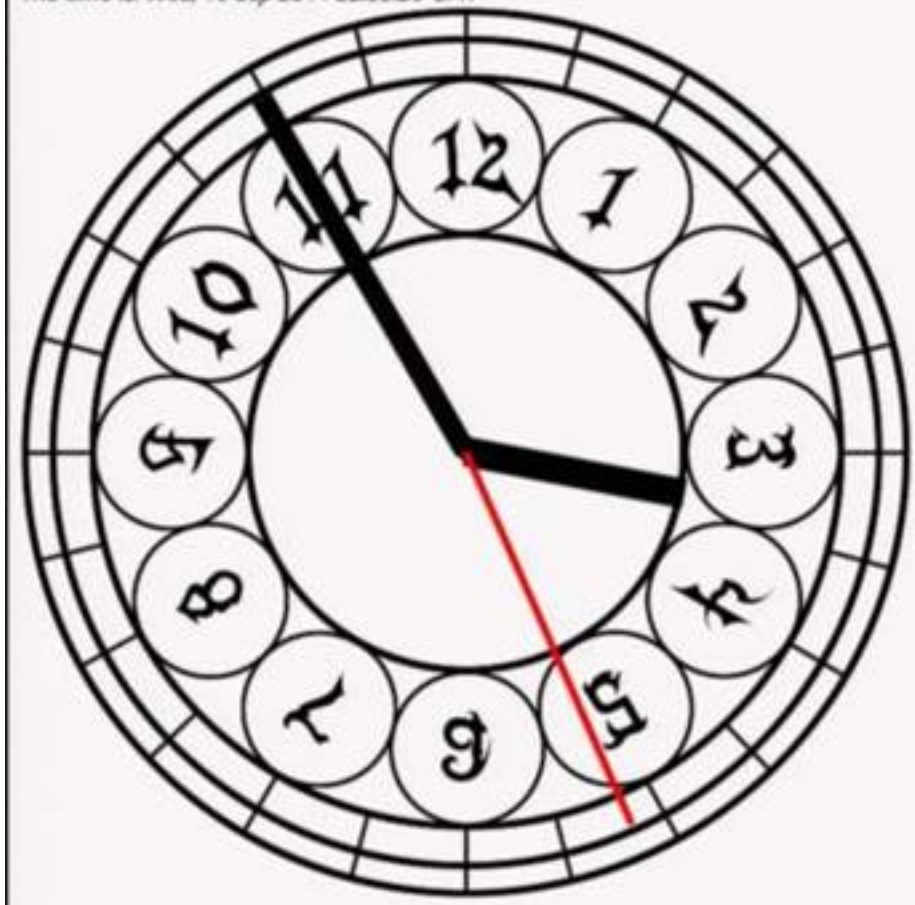
Location	Value
this	[object Window]
arguments	[object (Arguments)]
date	undefined
hours	undefined
minutes	undefined
seconds	undefined
[Scope]	ClockFace
[Scope]	[Anonymous function]
[Global]	
Add watch	

Call stack Breakpoints

Function	Location
Anonymous function	Clocks (33, 17)

Old Style Clock App

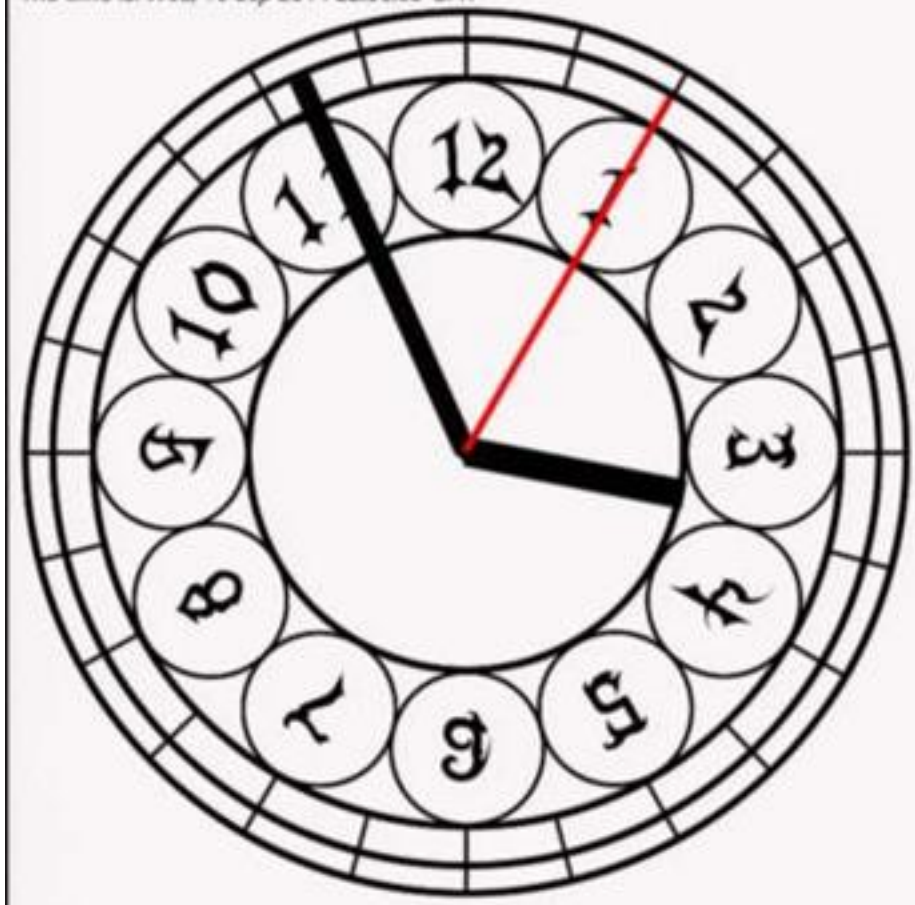
The time is: Wed, 10 Sep 2014 22:55:26 GMT




```
1  /// reference path="./Clock.js" />
2
3  module Clock {
4    export class App {
5      element: HTMLElement;
6      span: HTMLElement;
7      timerToken: number;
8      clock: Clock.ClockFace;
9
10     constructor(element: HTMLElement) {
11       this.element = element;
12       this.element.innerHTML += "The time is: ";
13       this.span = document.createElement('span');
14       this.element.appendChild(this.span);
15       this.span.innerText = new Date().toUTCString();
16
17       var clockDiv = document.createElement("div");
18       element.appendChild(clockDiv);
19       this.clock = new ClockFace(clockDiv);
20     }
21
22     start() {
23       this.timerToken = setInterval(() => this.span.innerHTML = new Date().toUTCString(), 1000);
24     }
25
26     stop() {
27       clearTimeout(this.timerToken);
28     }
29   }
30 }
31
32
33 window.onload = () => {
34   var el = document.getElementById('content');
35   var greeter = new Clock.App(el);
36   greeter.start();
37 };
```

Old Style Clock App

The time is: Wed, 10 Sep 2014 22:56:05 GMT



```
1  /// reference path="./Clock.ts" */
2
3  module Clock {
4      export class App {
5          element: HTMLElement;
6          span: HTMLElement;
7          timerToken: number;
8          clock: Clock.ClockFace;
9
10         constructor(element: HTMLElement) {
11             this.element = element;
12             this.element.innerHTML += "The time is: ";
13             this.span = document.createElement("span");
14             this.element.appendChild(this.span);
15             this.span.innerText = new Date().toUTCString();
16
17             var clockDiv = document.createElement("div");
18             element.appendChild(clockDiv);
19             this.clock = new ClockFace(clockDiv);
20         }
21
22         start() {
23             this.timerToken = setInterval(() => this.span.innerHTML = new Date().toUTCString(), 1000);
24         }
25
26         stop() {
27             clearInterval(this.timerToken);
28         }
29     }
30 }
31
32
33 window.onload = () => {
34     var el = document.getElementById("content");
35     var greeter = new Clock.App(el);
36     greeter.start();
37 };
```

this	[object App]
arguments	[object Arguments]
element	[object HTMLDivElement]
clockDiv	[object HTMLDivElement]
[Scope]	[Anonymous function]
[Globals]	
Add watch	

App	app.ts (19, 13)
window.onload	app.ts (35, 5)

Old Style Clock App

The time is: Wed, 10 Sep 2014 22:57:23 GMT



```
1  /// reference path="./Clock.js" />
2
3  module Clock {
4    export class App {
5      element: HTMLElement;
6      span: HTMLElement;
7      timerToken: number;
8      clock: Clock.ClockFace;
9
10     constructor(element: HTMLElement) {
11       this.element = element;
12       this.element.innerHTML += "The time is: ";
13       this.span = document.createElement("span");
14       this.element.appendChild(this.span);
15       this.span.innerHTML = new Date().toUTCString();
16
17       var clockDiv = document.createElement("div");
18       element.appendChild(clockDiv);
19       this.clock = new ClockFace(clockDiv);
20     }
21
22     start() {
23       this.timerToken = setInterval(() => this.span.innerHTML = new Date().toUTCString(), 1000);
24     }
25
26     stop() {
27       clearInterval(this.timerToken);
28     }
29   }
30 }
31
32
33 window.onload = () => {
34   var el = document.getElementById("content");
35   var greeter = new Clock.App(el);
36   greeter.start();
37 };
```

- ☒ app.js (14, 13)
- ☒ app.js (19, 13)


```
1 module Clock {
2   export class ClockFace {
3     private static Uri: string = "../img/clock1.png";
4     private static MinuteIncrement: number = 6;
5     private static HourIncrement: number = 30;
6
7     private element: HTMLDivElement;
8     private hourHand: HTMLElement;
9     private minuteHand: HTMLElement;
10    private secondHand: HTMLElement;
11
12    constructor(element: HTMLDivElement) {
13      var image: HTMLImageElement = document.createElement("img");
14      image.src = ClockFace.Uri;
15
16      element.classList.add("clockFace");
17      element.appendChild(image);
18
19      // Create hands
20      this.hourHand = document.createElement("div");
21      this.hourHand.classList.add("hourHand");
22      element.appendChild(this.hourHand);
23
24      this.minuteHand = document.createElement("div");
25      this.minuteHand.classList.add("minuteHand");
26      element.appendChild(this.minuteHand);
27
28      this.secondHand = document.createElement("div");
29      this.secondHand.classList.add("secondHand");
30      element.appendChild(this.secondHand);
31
32      setInterval(() => {
33        var date = new Date();
34
35        var hours = date.getHours() % 12;
36        var minutes = date.getMinutes();
37        var seconds = date.getSeconds();
38
39        this.hourHand.style.transform = "rotate(" + (hours * ClockFace.HourIncrement + (minutes * ClockFace.MinuteIncrement / 30)) + "deg)";
40        this.minuteHand.style.transform = "rotate(" + minutes * ClockFace.MinuteIncrement + "deg)";
41        this.secondHand.style.transform = "rotate(" + seconds * ClockFace.MinuteIncrement + "deg)";
42      }, 1000);
43    }
44  }
45 }
46 }
47 }
```

[Locals]	
this	[object Window]
arguments	[object (Arguments)]
date	undefined
hours	undefined
minutes	undefined
seconds	undefined
[Scope]	ClockFace
[Scope]	[Anonymous function]
[Global]	
Add watch	

Main frame	
Anonymous function	Clocks (33, 17)

default.htm app.js clock.js

```
1 var Clock;
2 (function (Clock) {
3     var ClockFace = (function () {
4         function ClockFace(element) {
5             var _this = this;
6             var image = document.createElement("img");
7             image.src = ClockFace.Url;
8
9             element.classList.add("clockFace");
10            element.appendChild(image);
11
12            // Create hands
13            this.hourHand = document.createElement("div");
14            this.hourHand.classList.add("hourHand");
15            element.appendChild(this.hourHand);
16
17            this.minuteHand = document.createElement("div");
18            this.minuteHand.classList.add("minuteHand");
19            element.appendChild(this.minuteHand);
20
21            this.secondHand = document.createElement("div");
22            this.secondHand.classList.add("secondHand");
23            element.appendChild(this.secondHand);
24
25            setInterval(function () {
26                var date = new Date();
27
28                var hours = date.getHours() % 12;
29                var minutes = date.getMinutes();
30                var seconds = date.getSeconds();
31
32                _this.hourHand.style.transform = "rotate(" + (hours * ClockFace.HourIncrement + (minutes * ClockFace.MinuteIncrement / 30)) + "deg)";
33                _this.minuteHand.style.transform = "rotate(" + minutes * ClockFace.MinuteIncrement + "deg)";
34                _this.secondHand.style.transform = "rotate(" + seconds * ClockFace.MinuteIncrement + "deg)";
35            }, 500);
36        }
37        ClockFace.Url = "../img/clock1.jpg";
38        ClockFace.MinuteIncrement = 6;
39        ClockFace.HourIncrement = 30;
40        return ClockFace;
41    })();
42    Clock.ClockFace = ClockFace;
43 })(Clock || (Clock = {}));
44 /** sourceMappingURL=Clock.js.map
45
```

Watches

Find in files (Ctrl+F)

- [Locals]
- this [object Window]
- arguments [object (Arguments)]
- date undefined
- hours undefined
- minutes undefined
- seconds undefined
- [Scope]
- this [object (ClockFace)]
- [Scope] [Anonymous function]
- [Globes]
- Add watch

Call stack Breakpoints

Main frame

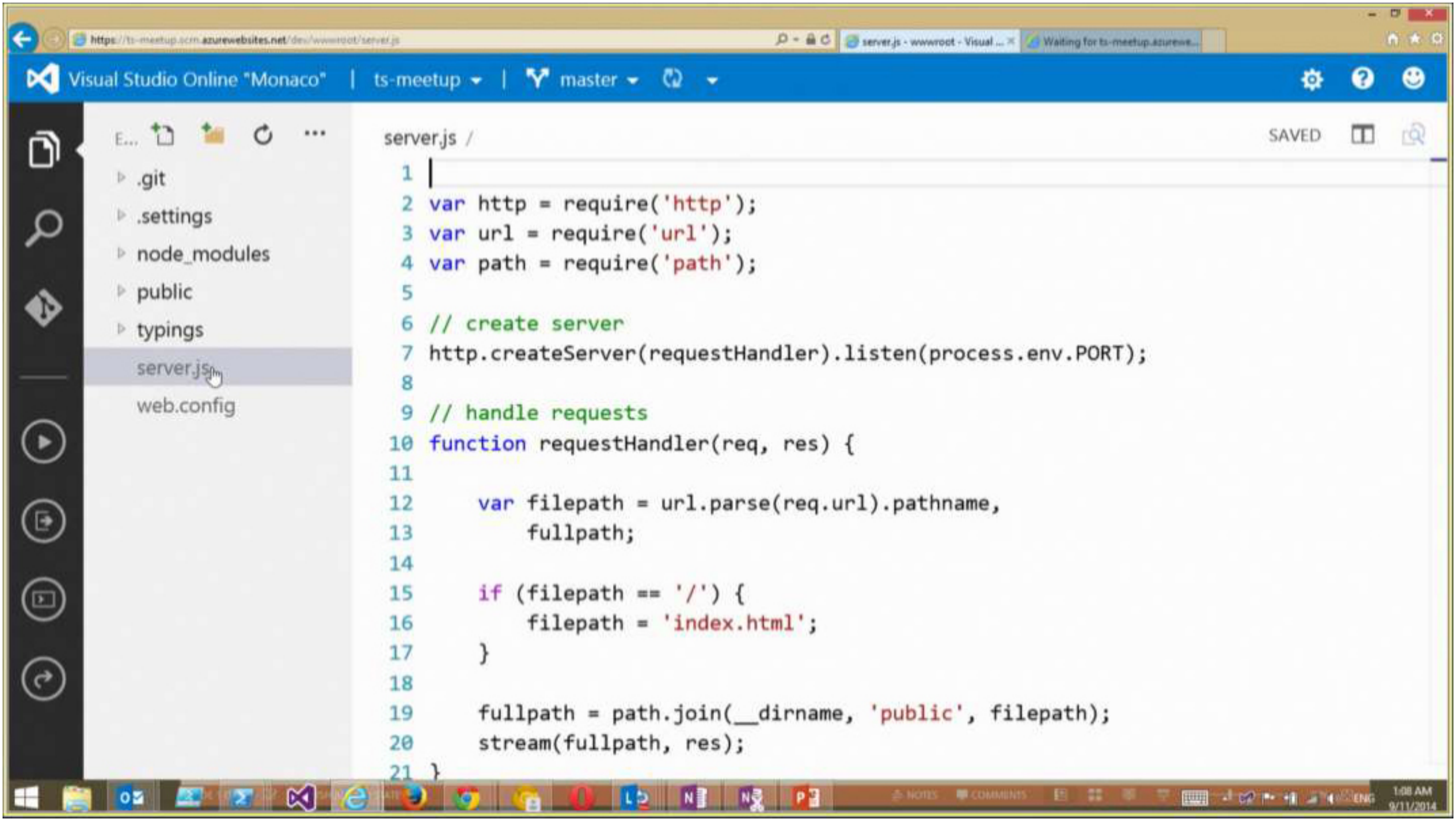
- Anonymous function clock.js (26, 17)

TypeScript – the man behind the curtain of the Monaco JavaScript tools

Erich Gamma, Johannes Rieken
VS Platform Tools/Monaco

»Chuck Norris can
solve the Towers of
Hanoi in one move.«





Visual Studio Online "Monaco"

ts-meetup > master



E... ...

- ▶ .git
- ▶ .settings
- ▶ node_modules
- ▶ public
- ▶ typings
- server.js
- web.config



server.js /

SAVED

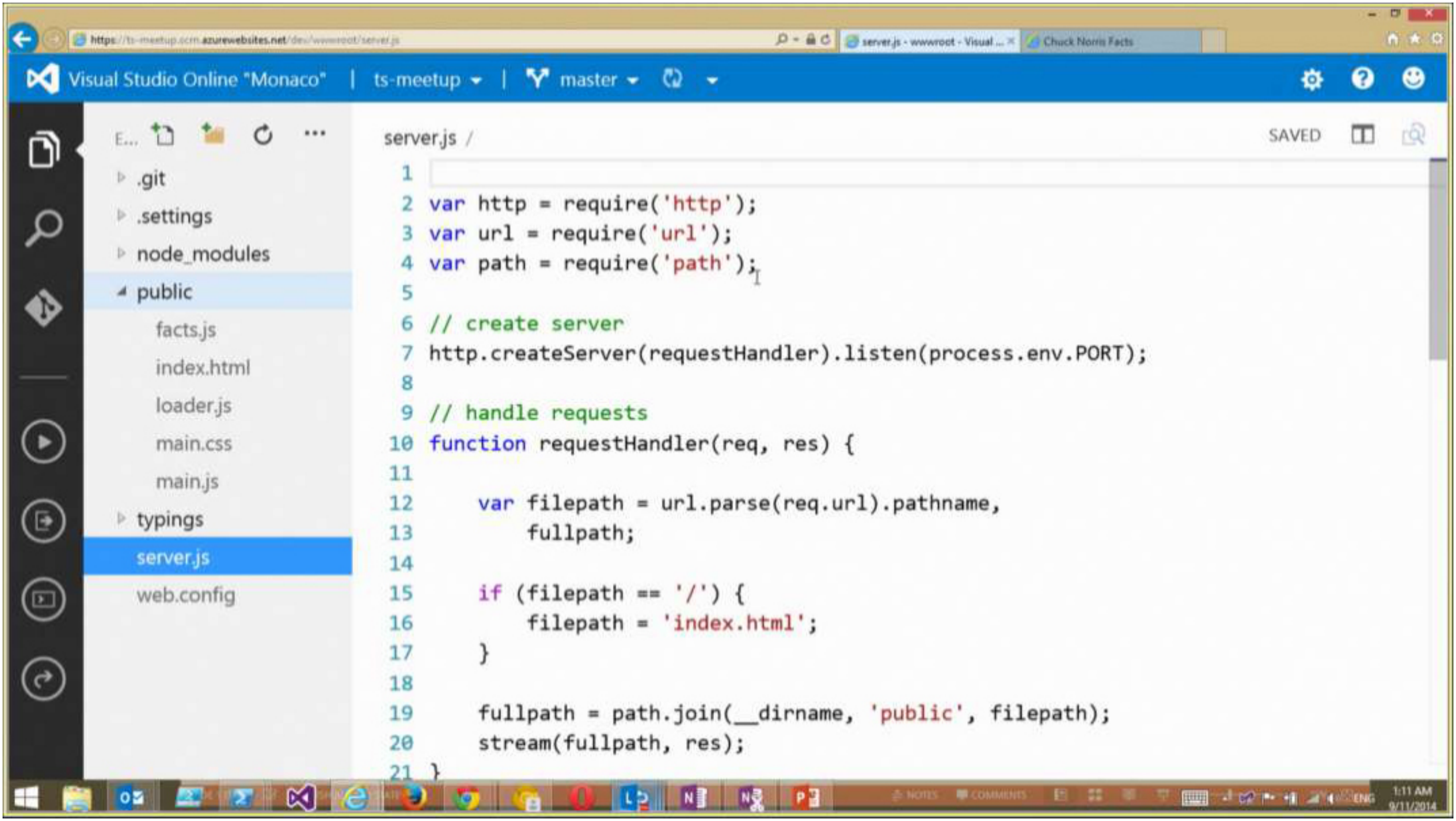
```
1 |
2 var http = require('http');
3 var url = require('url');
4 var path = require('path');
5
6 // create server
7 http.createServer(requestHandler).listen(process.env.PORT);
8
9 // handle requests
10 function requestHandler(req, res) {
11
12     var filepath = url.parse(req.url).pathname,
13         fullpath;
14
15     if (filepath == '/') {
16         filepath = 'index.html';
17     }
18
19     fullpath = path.join(__dirname, 'public', filepath);
20     stream(fullpath, res);
21 }
```


»Chuck Norris doesn't
use web standards as
the web will conform
to him.«

» Whiteboards are
white because Chuck
Norris scared them
that way.«

»Chuck Norris burst
the dot com bubble.«

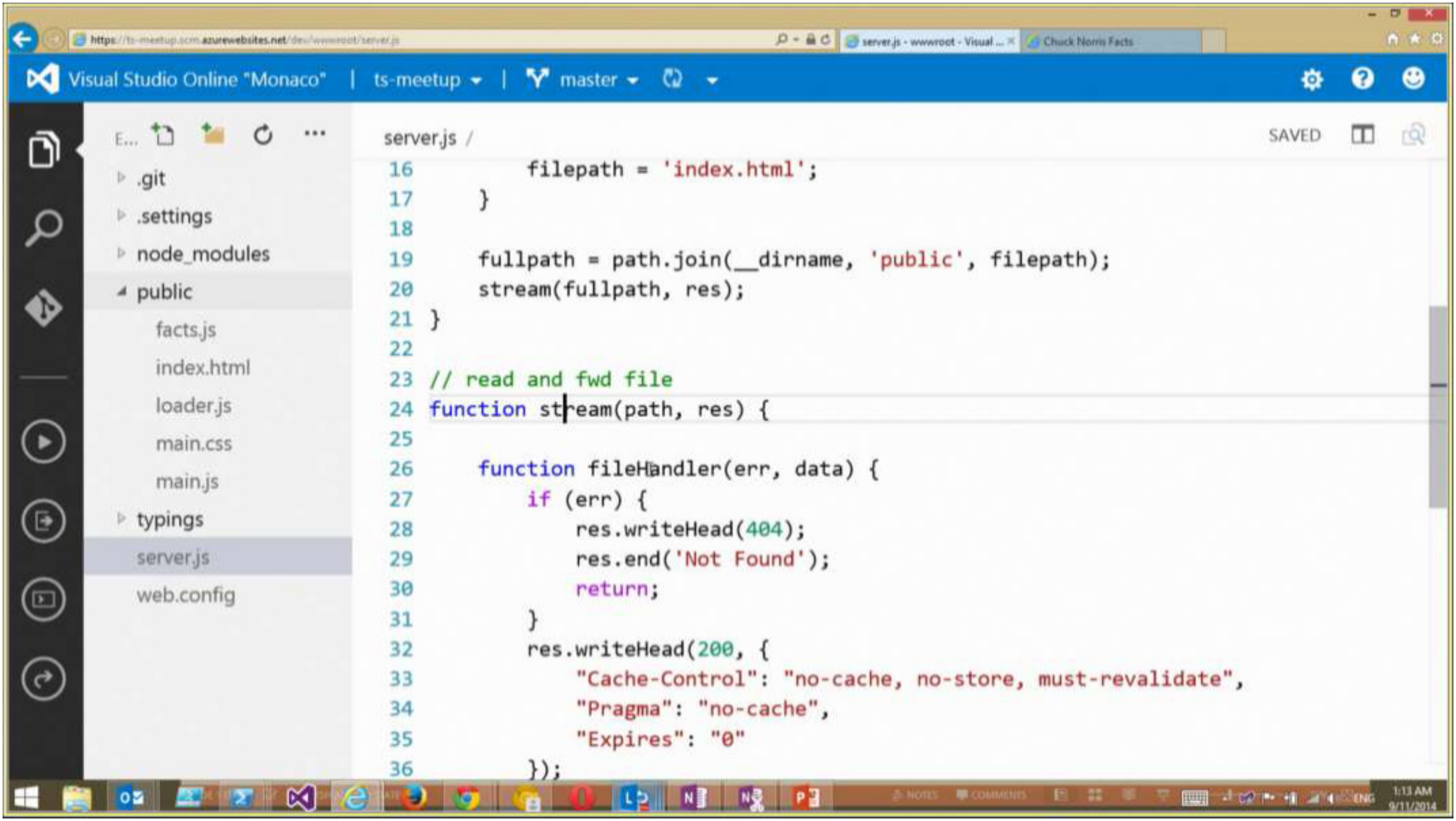


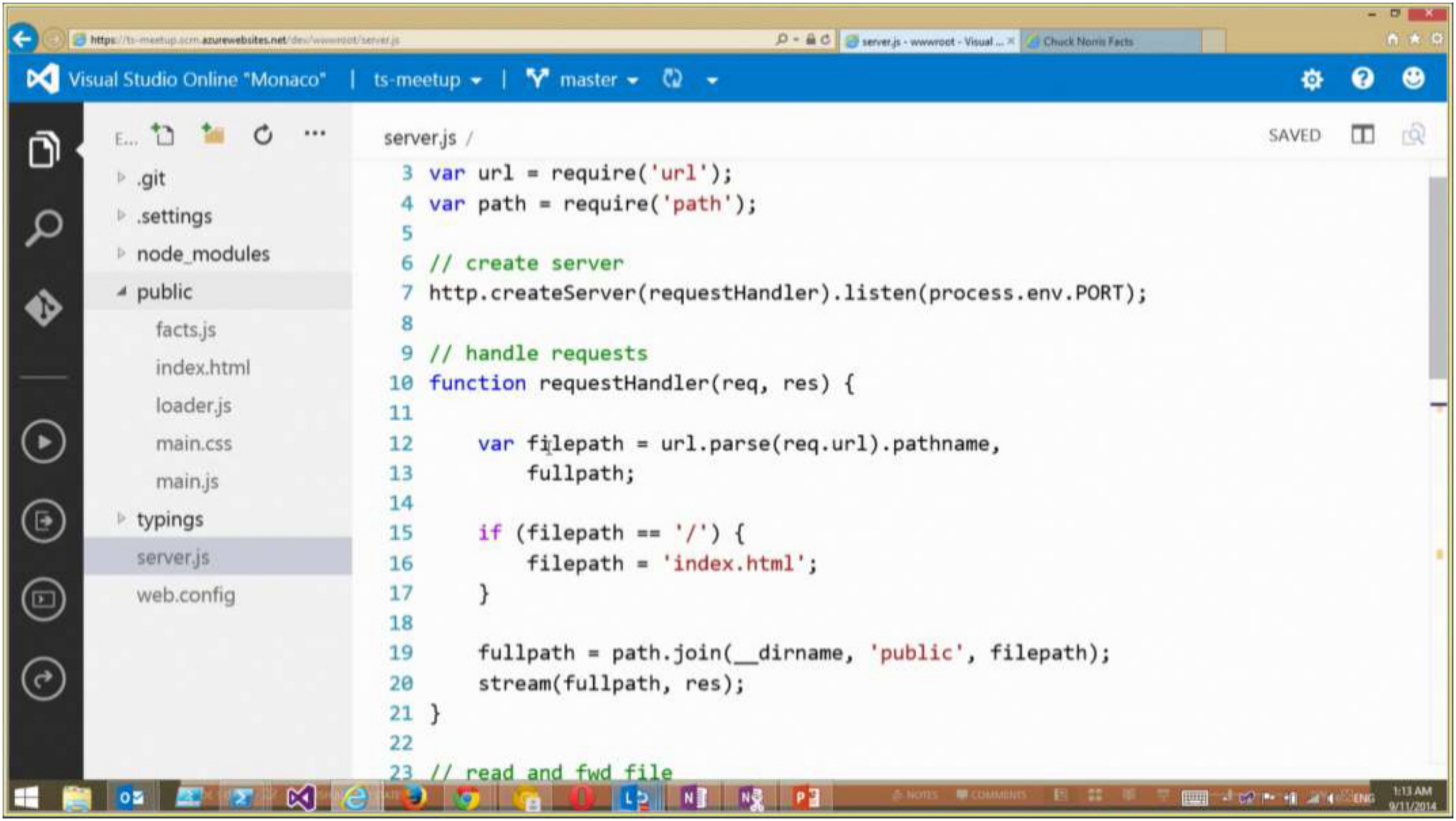


- E...
- .git
- .settings
- node_modules
- public
 - facts.js
 - index.html
 - loader.js
 - main.css
 - main.js
- typings
- server.js
- web.config

server.js /

```
1
2 var http = require('http');
3 var url = require('url');
4 var path = require('path');
5
6 // create server
7 http.createServer(requestHandler).listen(process.env.PORT);
8
9 // handle requests
10 function requestHandler(req, res) {
11
12     var filepath = url.parse(req.url).pathname,
13         fullpath;
14
15     if (filepath == '/') {
16         filepath = 'index.html';
17     }
18
19     fullpath = path.join(__dirname, 'public', filepath);
20     stream(fullpath, res);
21 }
```



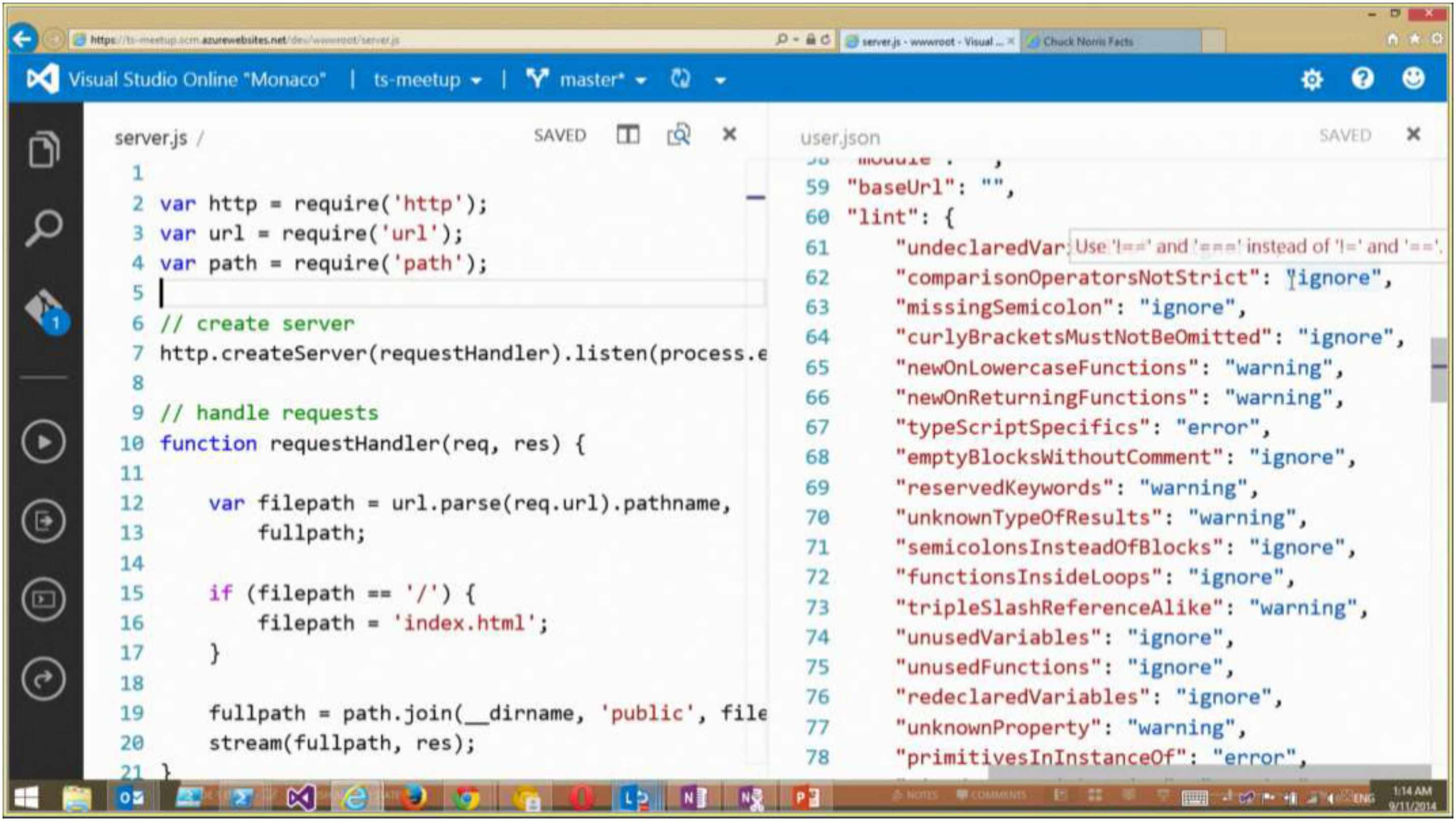


- ...
- ▶ .git
- ▶ .settings
- ▶ node_modules
- ▶ public
 - facts.js
 - index.html
 - loader.js
 - main.css
 - main.js
- ▶ typings
- server.js
- web.config

server.js /

SAVED

```
3 var url = require('url');
4 var path = require('path');
5
6 // create server
7 http.createServer(requestHandler).listen(process.env.PORT);
8
9 // handle requests
10 function requestHandler(req, res) {
11
12     var filepath = url.parse(req.url).pathname,
13         fullpath;
14
15     if (filepath == '/') {
16         filepath = 'index.html';
17     }
18
19     fullpath = path.join(__dirname, 'public', filepath);
20     stream(fullpath, res);
21 }
22
23 // read and fwd file
```



server.js /

SAVED



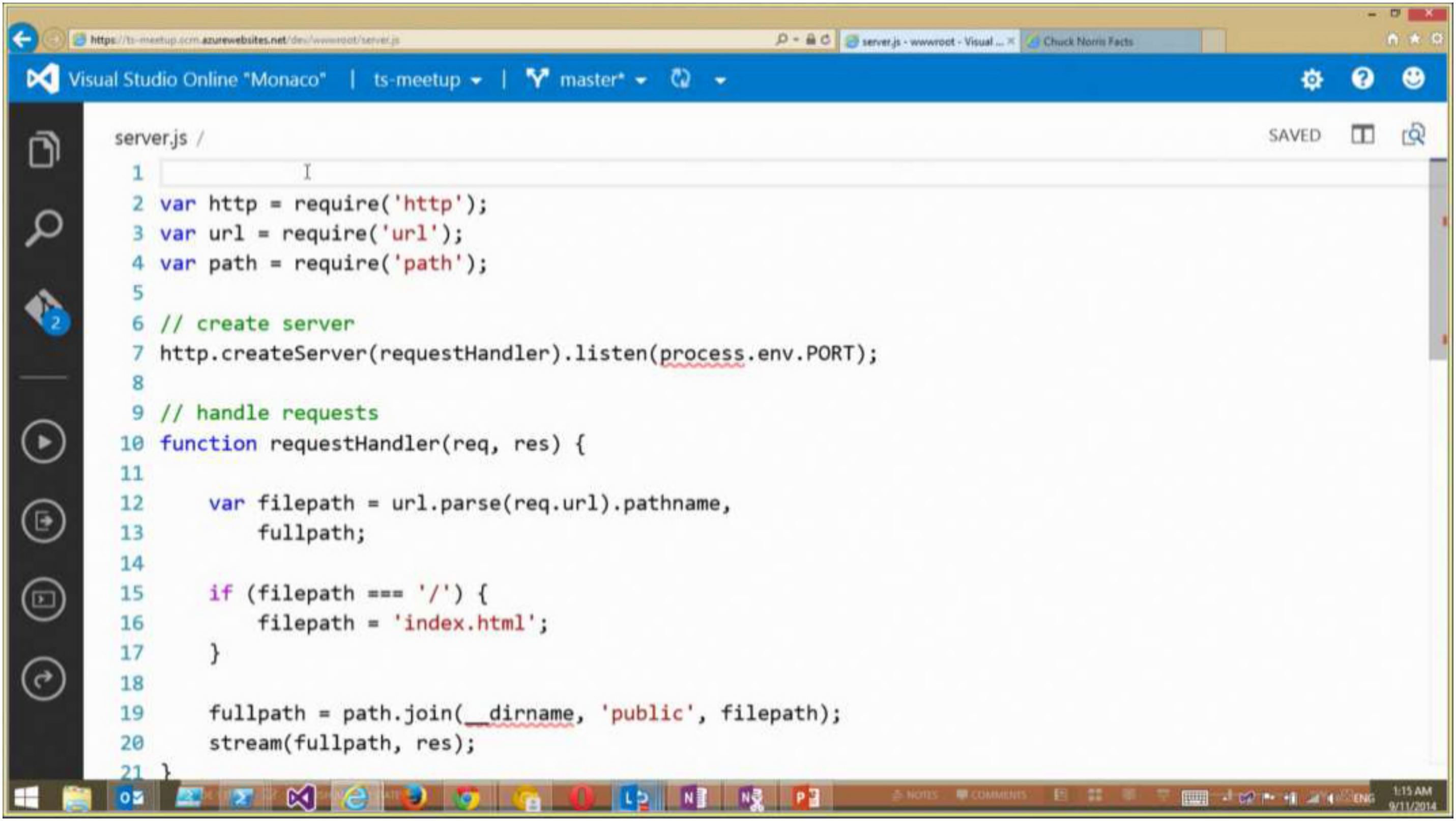
user.json

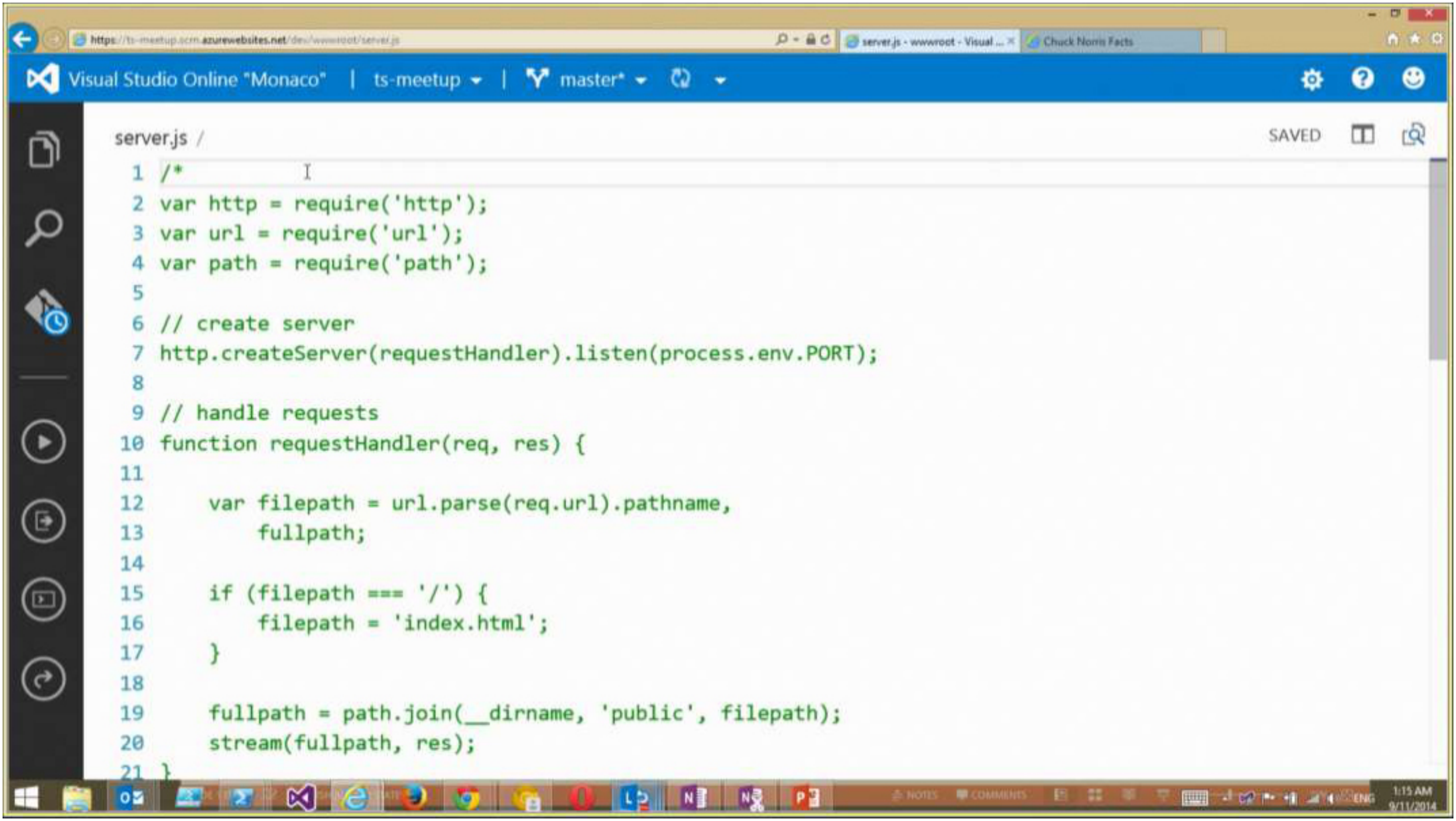
SAVED

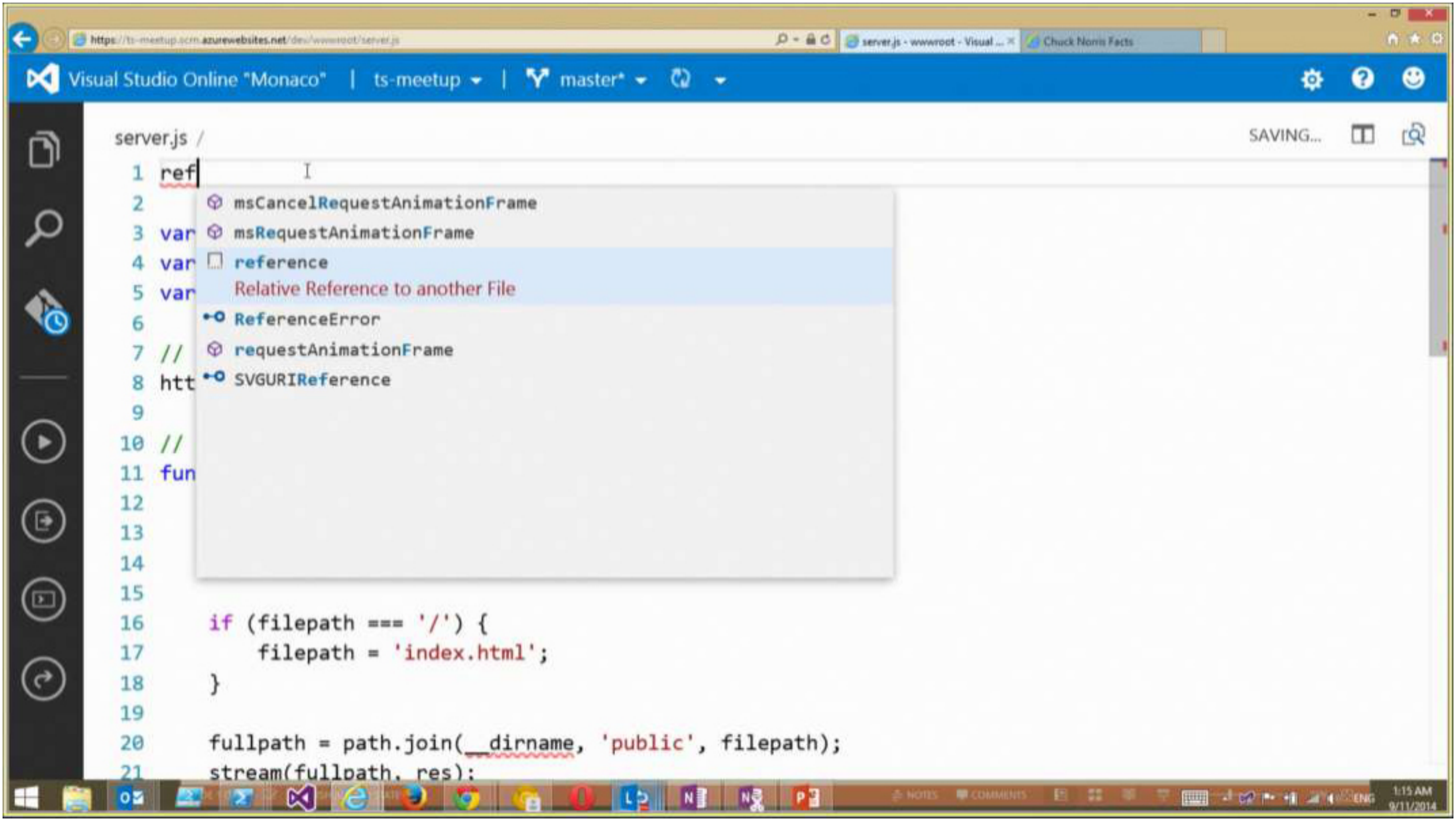


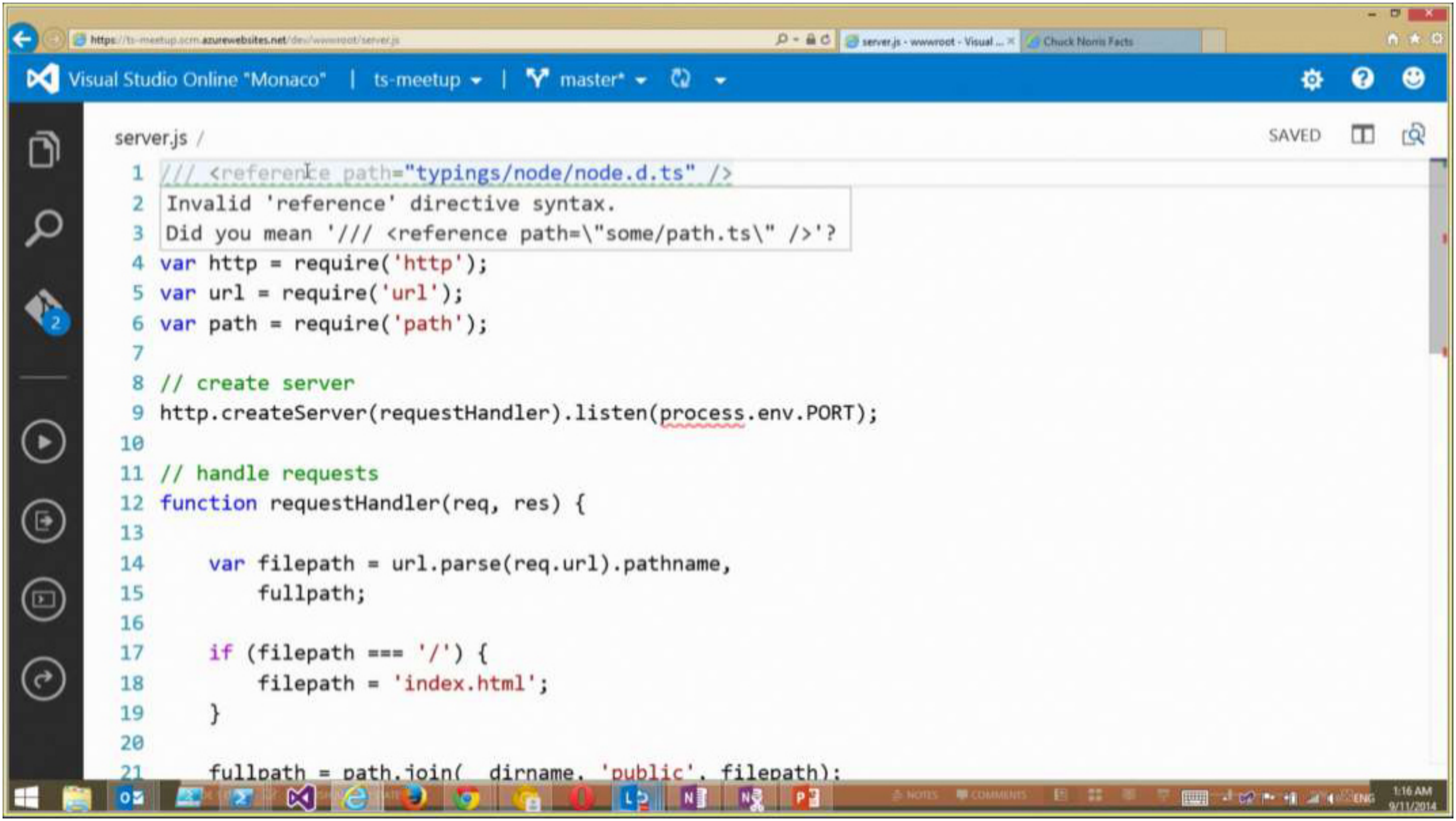
```
1
2 var http = require('http');
3 var url = require('url');
4 var path = require('path');
5
6 // create server
7 http.createServer(requestHandler).listen(process.e
8
9 // handle requests
10 function requestHandler(req, res) {
11
12     var filepath = url.parse(req.url).pathname,
13         fullpath;
14
15     if (filepath == '/') {
16         filepath = 'index.html';
17     }
18
19     fullpath = path.join(__dirname, 'public', file
20     stream(fullpath, res);
21 }
```

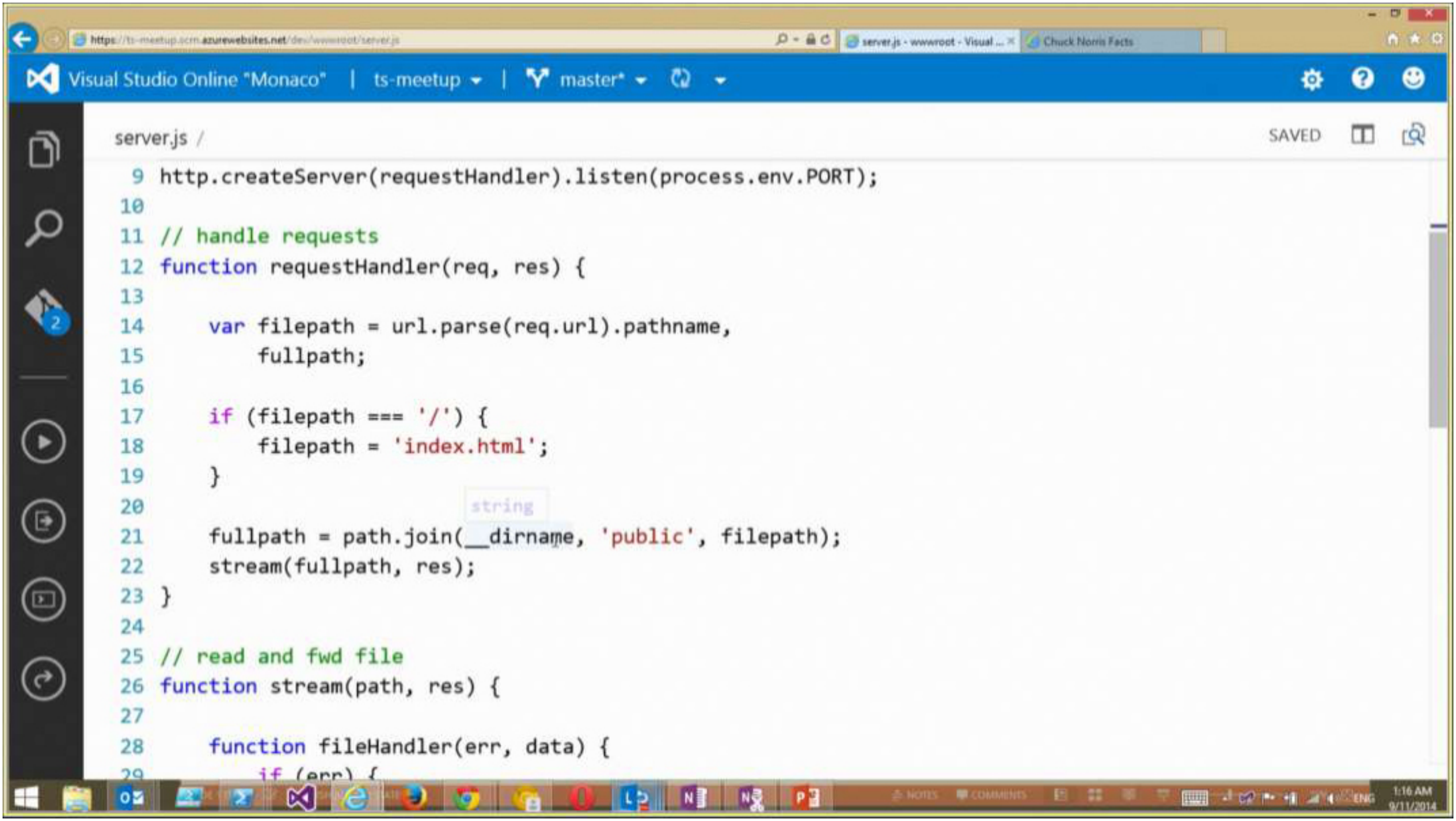
```
58 module .
59 "baseUrl": "",
60 "lint": {
61     "undeclaredVar: Use '===' and '===' instead of '!=' and '=='.
62     "comparisonOperatorsNotStrict": "ignore",
63     "missingSemicolon": "ignore",
64     "curlyBracketsMustNotBeOmitted": "ignore",
65     "newOnLowercaseFunctions": "warning",
66     "newOnReturningFunctions": "warning",
67     "typeScriptSpecifics": "error",
68     "emptyBlocksWithoutComment": "ignore",
69     "reservedKeywords": "warning",
70     "unknownTypeOfResults": "warning",
71     "semicolonsInsteadOfBlocks": "ignore",
72     "functionsInsideLoops": "ignore",
73     "tripleSlashReferenceAlike": "warning",
74     "unusedVariables": "ignore",
75     "unusedFunctions": "ignore",
76     "redeclaredVariables": "ignore",
77     "unknownProperty": "warning",
78     "primitivesInInstanceOf": "error",
```

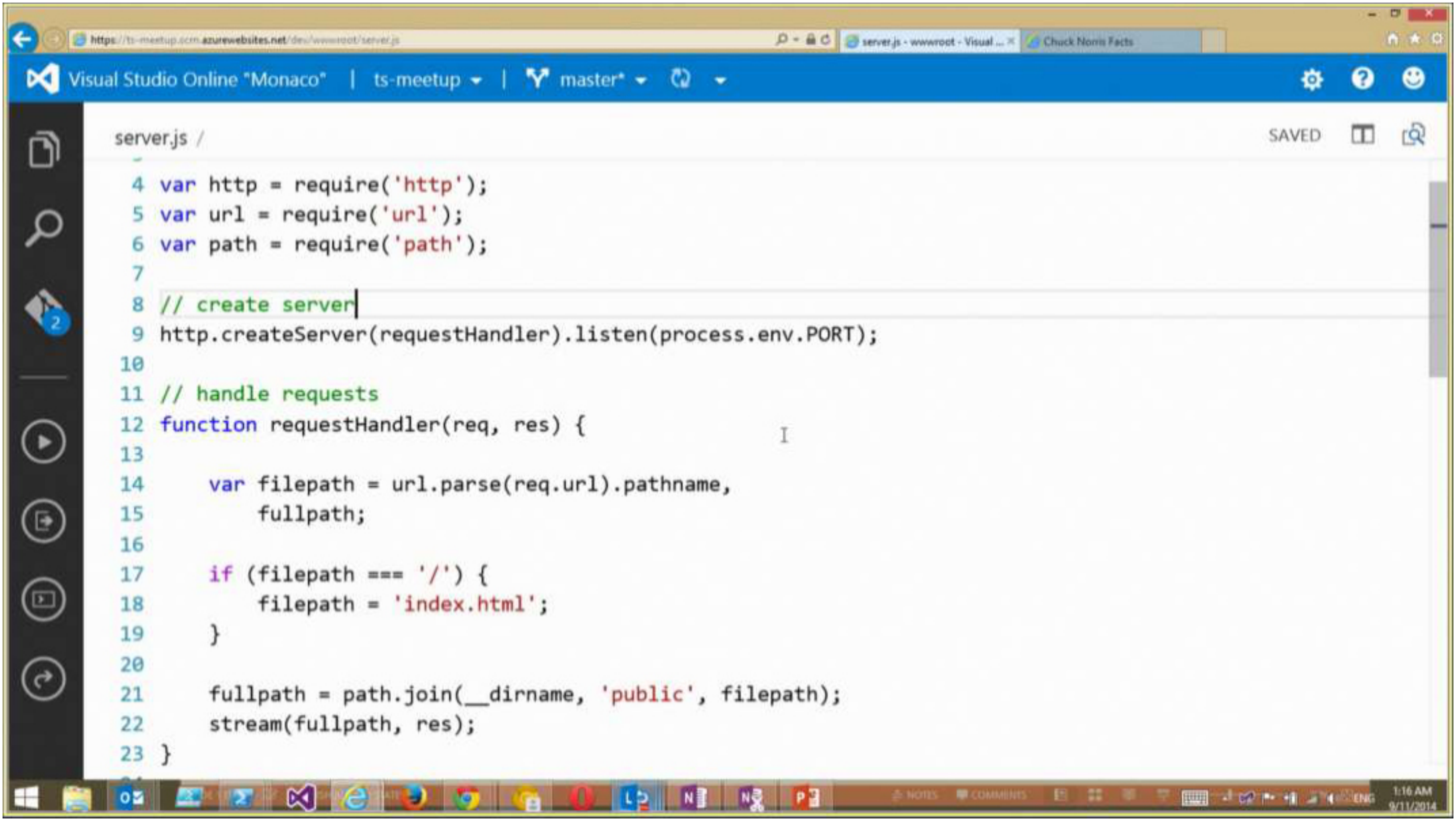



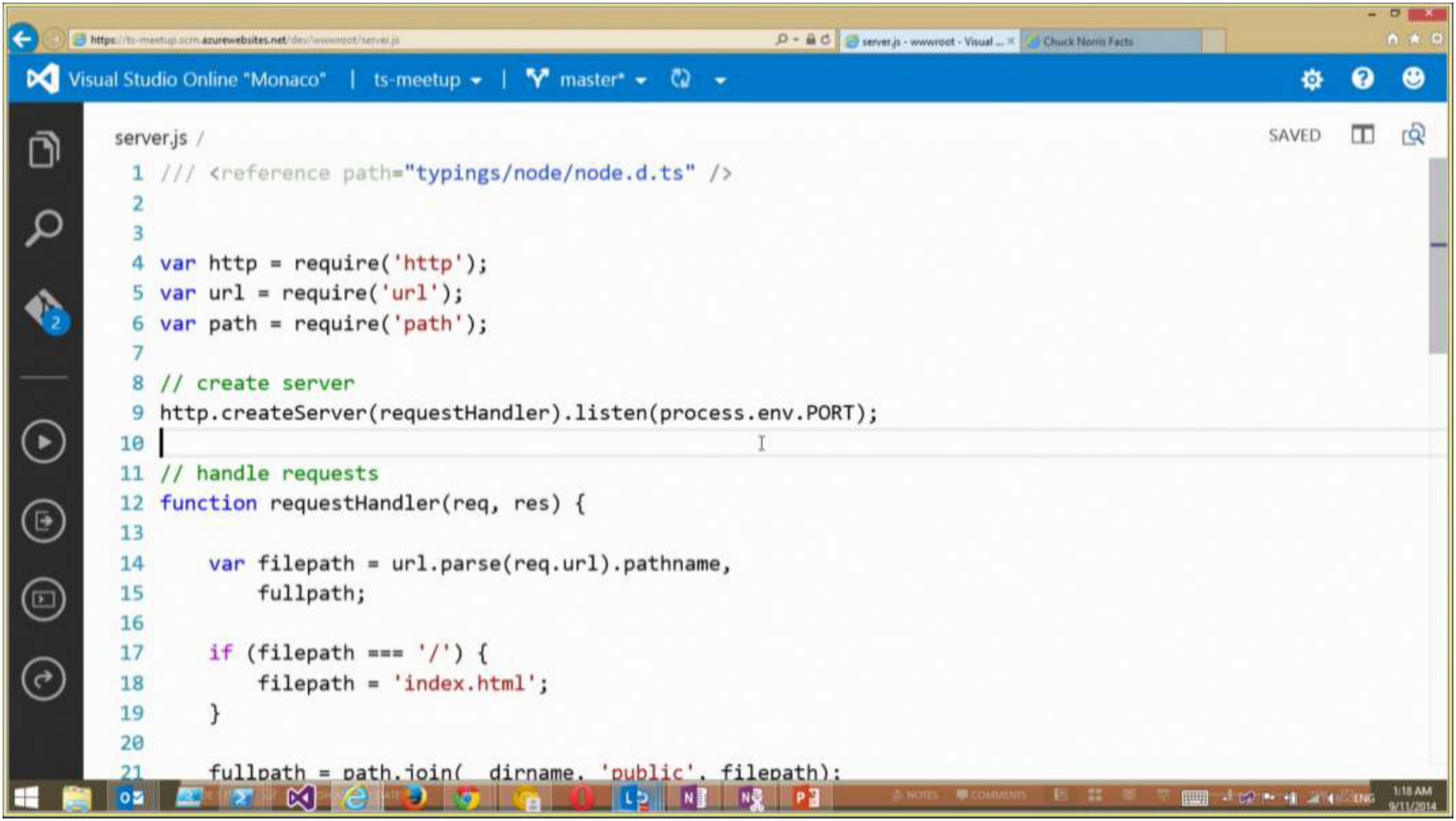












```

20
21     fullpath = path.join(__dirname, 'public', filepath);
22     stream(fullpath, res);
23 }
24
25 // read and fwd file
26 function stream(path, res) {
27
28     function fileHandler(err, data) {
29         if (err) {
30             res.writeHead(404);
31             res.end('Not Found');
32             return;
33         }
34         res.writeHead(200, {
35             "Cache-Control": "no-cache, no-store, must-revalidate",
36             "Pragma": "no-cache",
37             "Expires": "0"
38         });
39         res.write(data.toString());
40         res.end();
41     };
42
43     require('fs').readFile(path, fileHandler);
44 }
    
```

```

1  /// <reference path="typings/node/node.d.ts" />
2
3
4  declare var exports:any; declare var module:any; declare var __dirname:string;
5  import http = require('http');
6  import url = require('url');
7  import path = require('path');
8  import fs = require('fs');
9  var http = http;
10 var url = url;
11 var path = path;
12
13 // create server
14 http.createServer(requestHandler).listen(process.env.PORT);
15
16 // handle requests
17 function requestHandler(req, res) {
18
19     var filepath = url.parse(req.url).pathname,
20         fullpath;
21
22     if (filepath === '/') {
23         filepath = 'index.html';
24     }
25
26     fullpath = path.join(__dirname, 'public', filepath);
    
```



```

1  /// <reference path="typings/node/node.d.ts" />
2
3
4  var http = require('http');
5  var url = require('url');
6  var path = require('path');
7
8  // create server
9  http.createServer(requestHandler).listen(process.env.PORT);
10
11 // handle requests
12 function requestHandler(req, res) {
13
14     var filepath = url.parse(req.url).pathname,
15         fullpath;
16
17     if (filepath === '/') {
18         filepath = 'index.html';
19     }
20
21     fullpath = path.join(__dirname, 'public', filepath);
22     stream(fullpath, res);
23 }
24
25 // read and fwd file
26 function stream(path, res) {

```

```

1  /// <reference path="typings/node/node.d.ts" />
2
3
4  declare var exports:any; declare var module:any; declare var
5  import http = require('http');
6  import url = require('url');
7  import path = require('path');
8  import fs = require('fs');
9  var http = http;
10 var url = url;
11 var path = path;
12
13 // create server
14 http.createServer(requestHandler).listen(process.env.PORT);
15
16 // handle requests
17 function requestHandler(req, res) {
18
19     var filepath = url.parse(req.url).pathname,
20         fullpath;
21
22     if (filepath === '/') {
23         filepath = 'index.html';
24     }
25
26     fullpath = path.join(__dirname, 'public', filepath);

```

```

1  /// <reference path="typings/node/node.d.ts" />
2
3
4  var http = require('http');
5  var url = require('url');
6  var path = require('path');
7
8  // create server
9  http.createServer(requestHandler).listen(process.env.PORT);
10
11 // handle requests
12 function requestHandler(req, res) {
13
14     var filepath = url.parse(req.url).pathname,
15         fullpath;
16
17     if (filepath === '/') {
18         filepath = 'index.html';
19     }
20
21     fullpath = path.join(__dirname, 'public', filepath);
22     stream(fullpath, res);
23 }
24
25 // read and fwd file
26 function stream(path, res) {

```

```

6  import url = require('url');
7  import path = require('path');
8  import fs = require('fs');
9  var http = http;
10 var url = url;
11 var path = path;
12
13 // create server
14 http.createServer(requestHandler).listen(process.env.PORT);
15
16 // handle requests
17 function requestHandler(req, res) {
18
19     var filepath = url.parse(req.url).pathname,
20         fullpath;
21
22     if (filepath === '/') {
23         filepath = 'index.html';
24     }
25
26     fullpath = path.join(__dirname, 'public', filepath);
27     stream(fullpath, res);
28 }
29
30 // read and fwd file
31 function stream(path, res) {
32

```



```

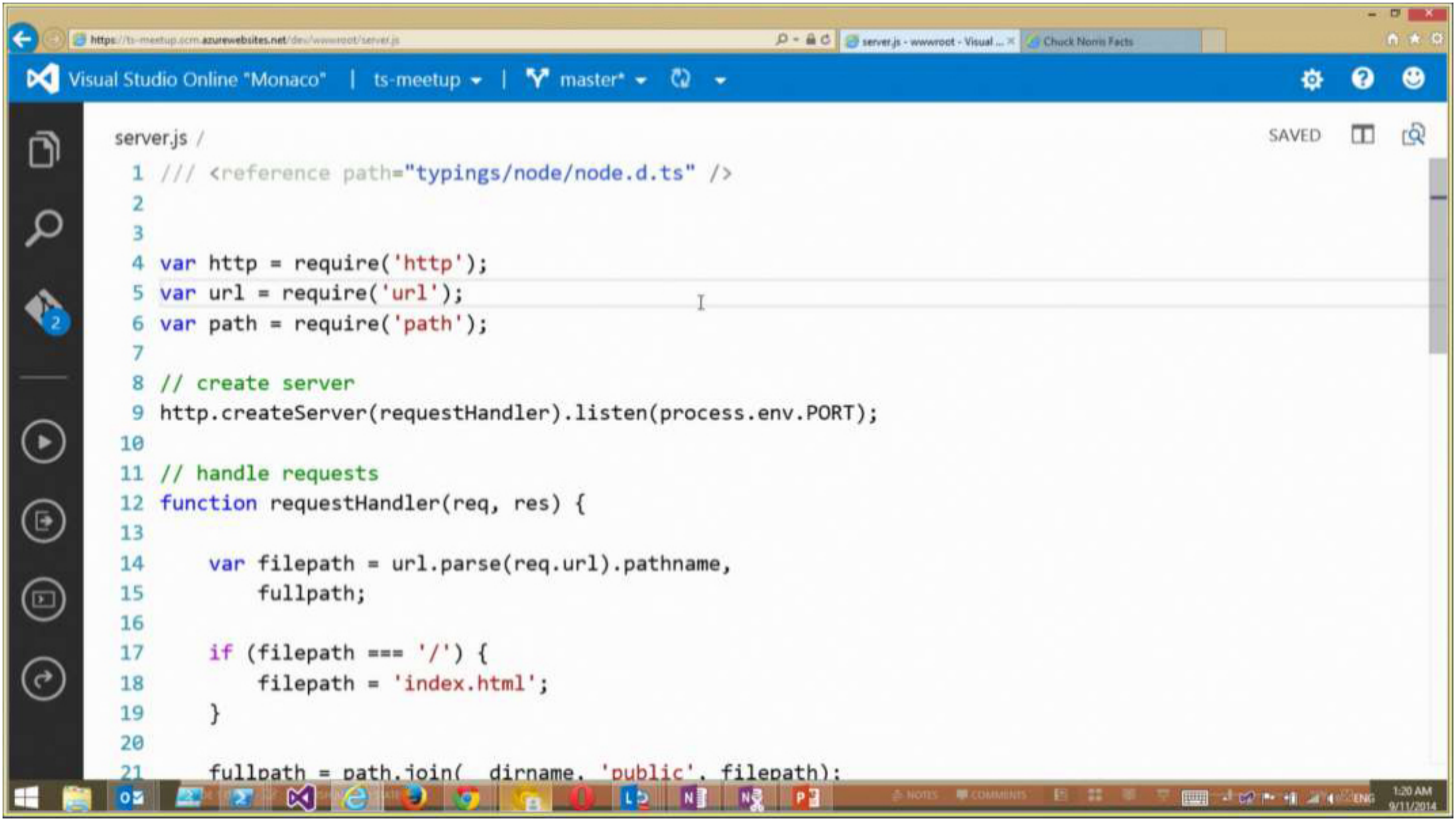
1  /// <reference path="typings/node/node.d.ts" />
2
3
4  var http = require('http');
5  var url = require('url');
6  var path = require('path');
7
8  // create server
9  http.createServer(requestHandler).listen(process.env.PORT);
10
11 // handle requests
12 function requestHandler(req, res) {
13
14     var filepath = url.parse(req.url).pathname,
15         fullpath;
16
17     if (filepath === '/') {
18         filepath = 'index.html';
19     }
20
21     fullpath = path.join(__dirname, 'public', filepath);
22     stream(fullpath, res);
23 }
24
25 // read and fwd file
26 function stream(path, res) {

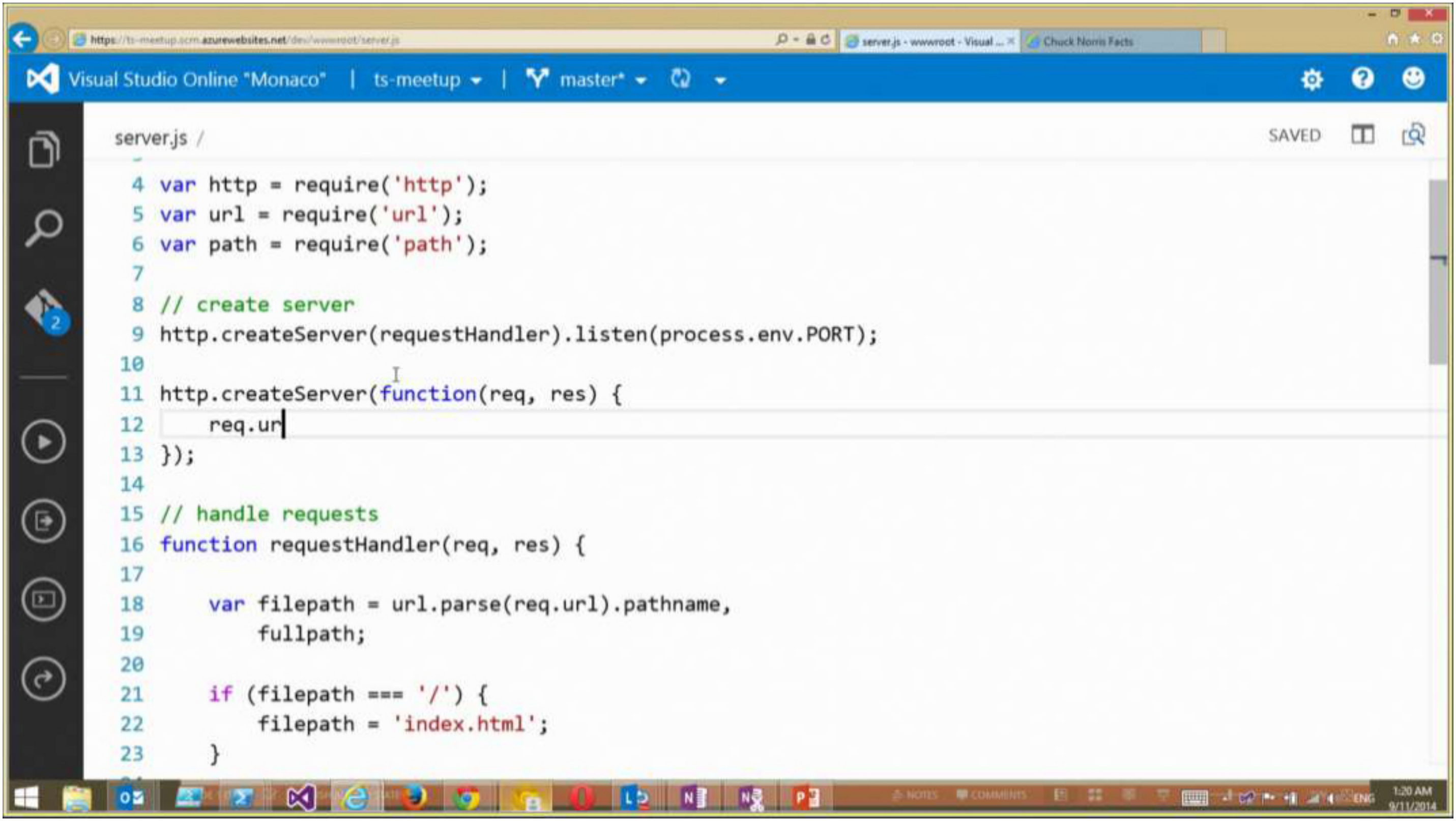
```

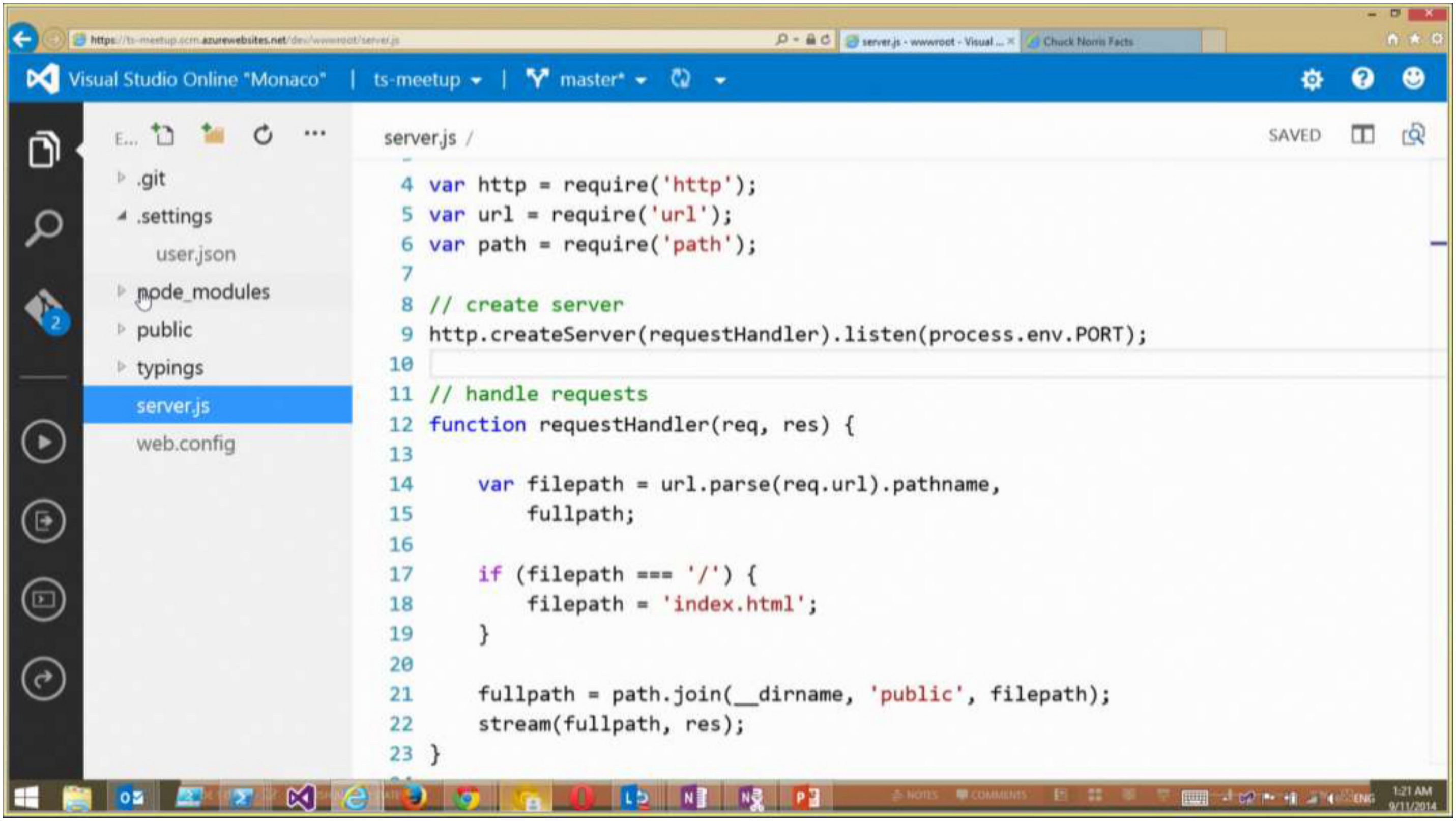
```

9  var http = http;
10 var url = url;
11 var path = path;
12
13 // create server
14 http.createServer(requestHandler).listen(process.env.PORT);
15
16 // handle requests
17 function requestHandler(req, res) {
18
19     var filepath = url.parse(req.url).pathname,
20         fullpath;
21
22     if (filepath === '/') {
23         filepath = 'index.html';
24     }
25
26     fullpath = path.join(__dirname, 'public', filepath);
27     stream(fullpath, res);
28 }
29
30 // read and fwd file
31 function stream(path, res) {
32
33     function fileHandler(err, data) {
34         if (err) {

```







- ...
- .git
- .settings
 - user.json
- node_modules
- public
- typings
- server.js**
- web.config

server.js /

SAVED

```
4 var http = require('http');
5 var url = require('url');
6 var path = require('path');
7
8 // create server
9 http.createServer(requestHandler).listen(process.env.PORT);
10
11 // handle requests
12 function requestHandler(req, res) {
13
14     var filepath = url.parse(req.url).pathname,
15         fullpath;
16
17     if (filepath === '/') {
18         filepath = 'index.html';
19     }
20
21     fullpath = path.join(__dirname, 'public', filepath);
22     stream(fullpath, res);
23 }
```


»Chuck Norris's beard
can type 140 wpm.«

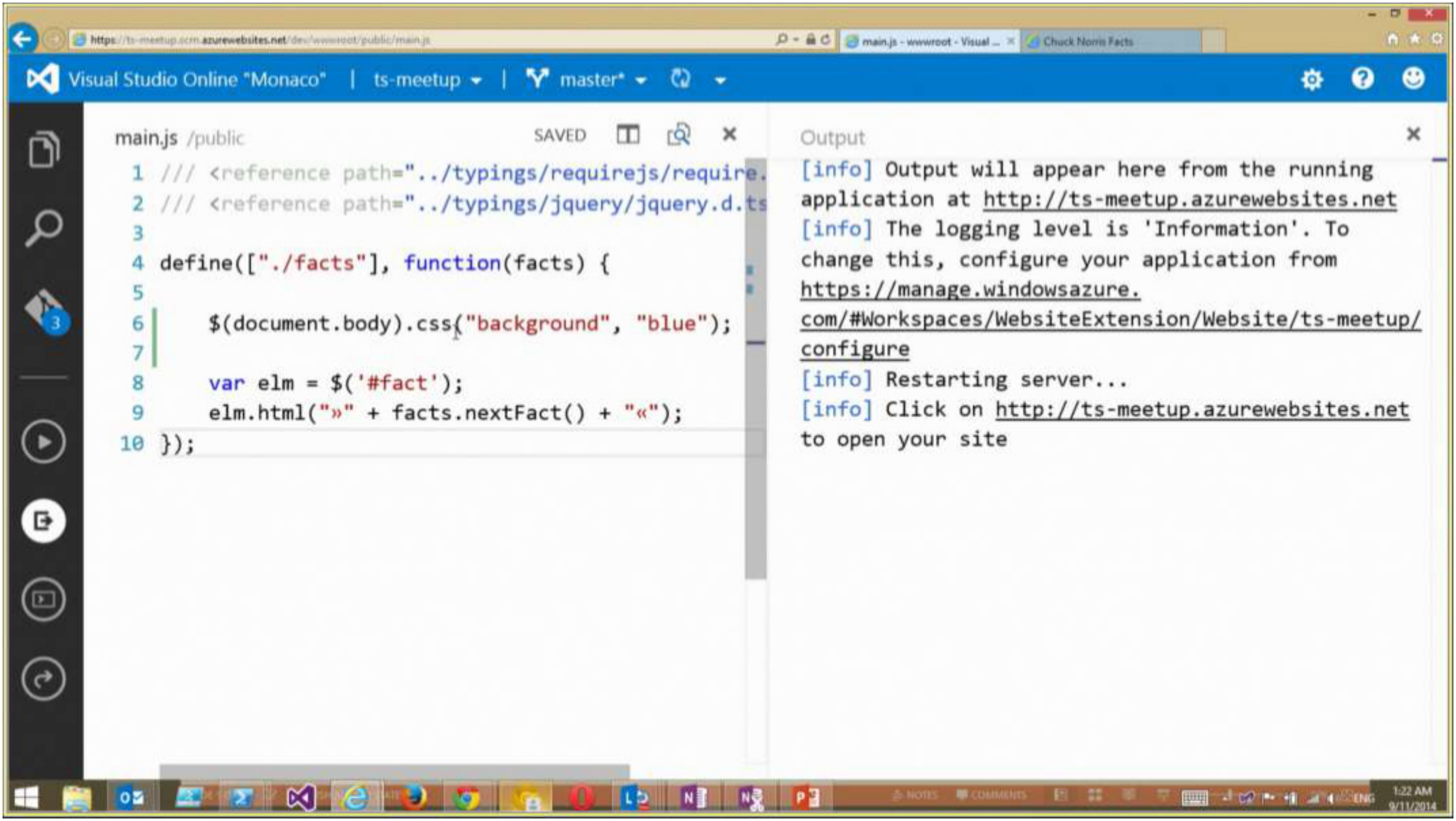


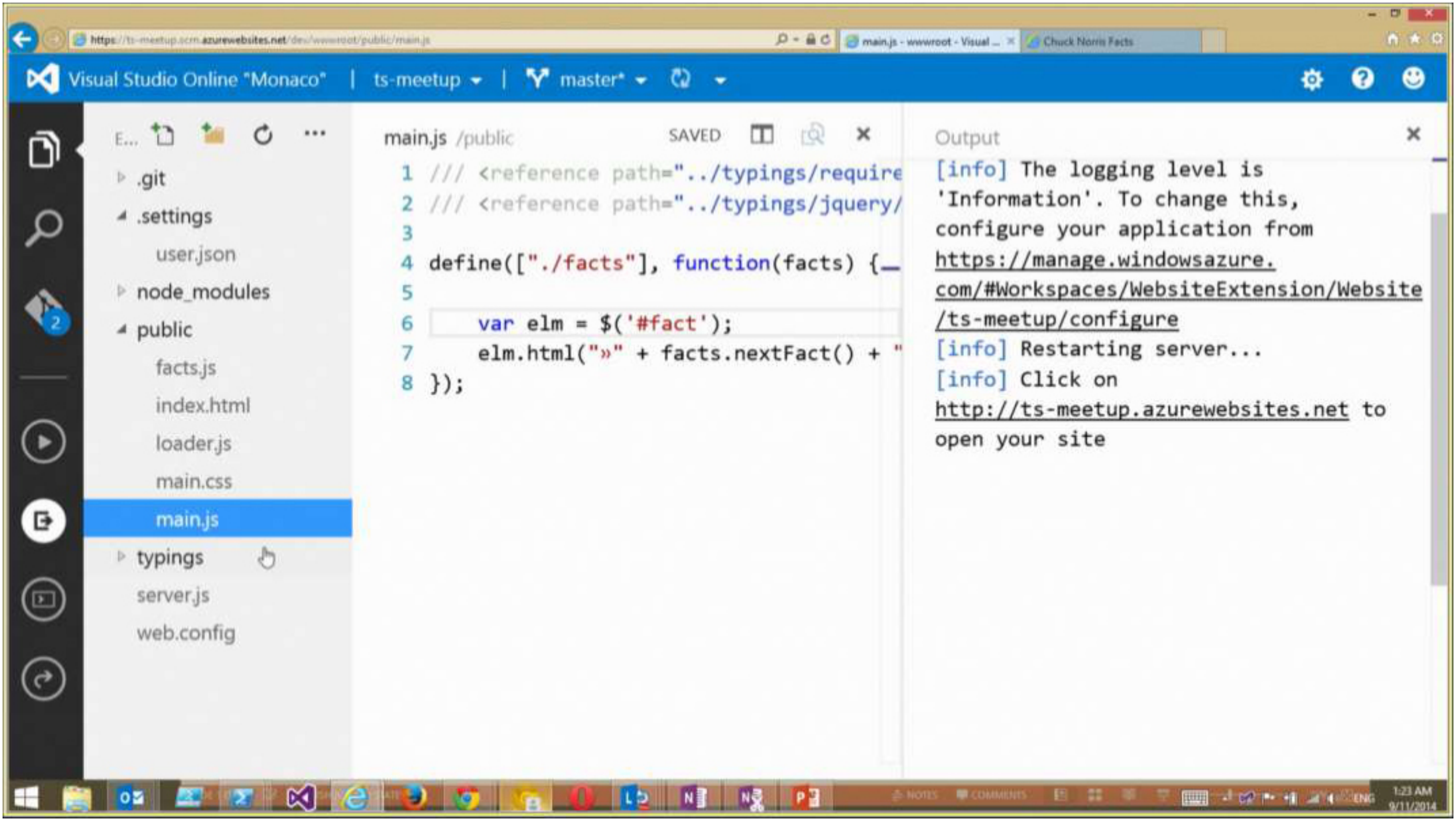
»»“It works on my
machine” always holds
true for Chuck
Norris.«

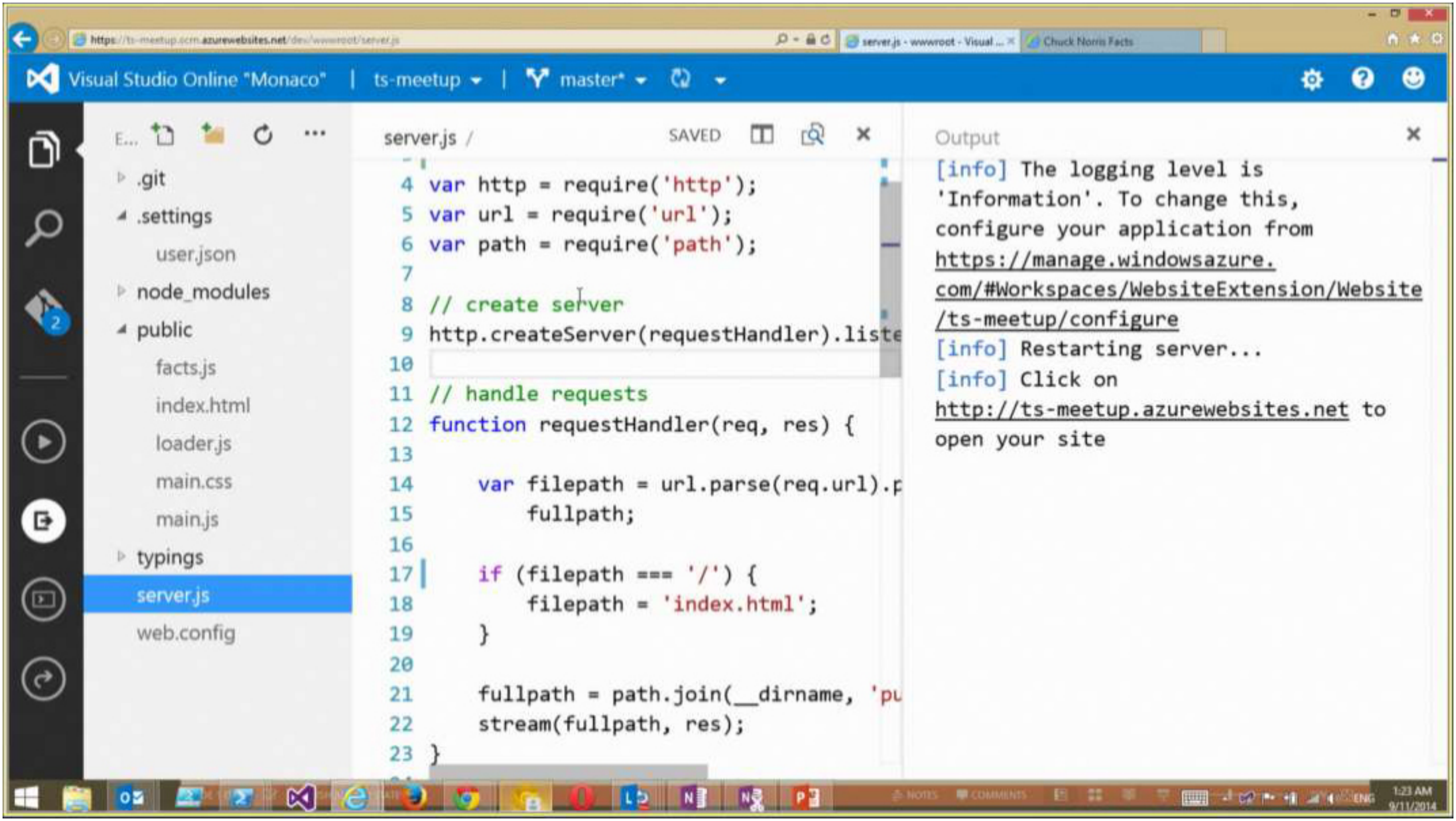


»Chuck Norris burst
the dot com bubble.«









Visual Studio Online "Monaco"

ts-meetup

master



E... ...

▶ .git

▶ .settings

user.json

▶ node_modules

▶ public

facts.js

index.html

loader.js

main.css

main.js

▶ typings

server.js

web.config

server.js /

SAVED



Output

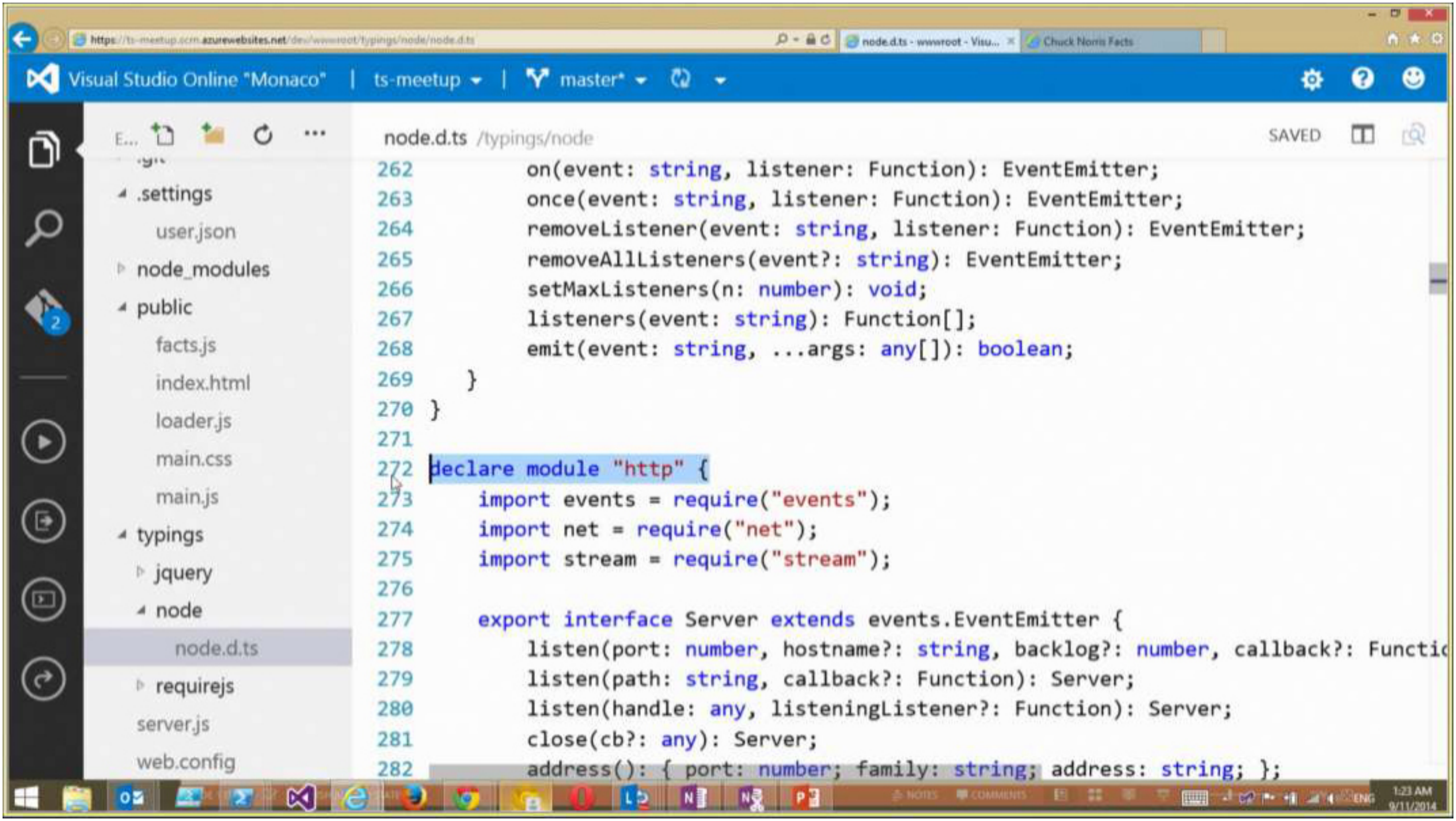


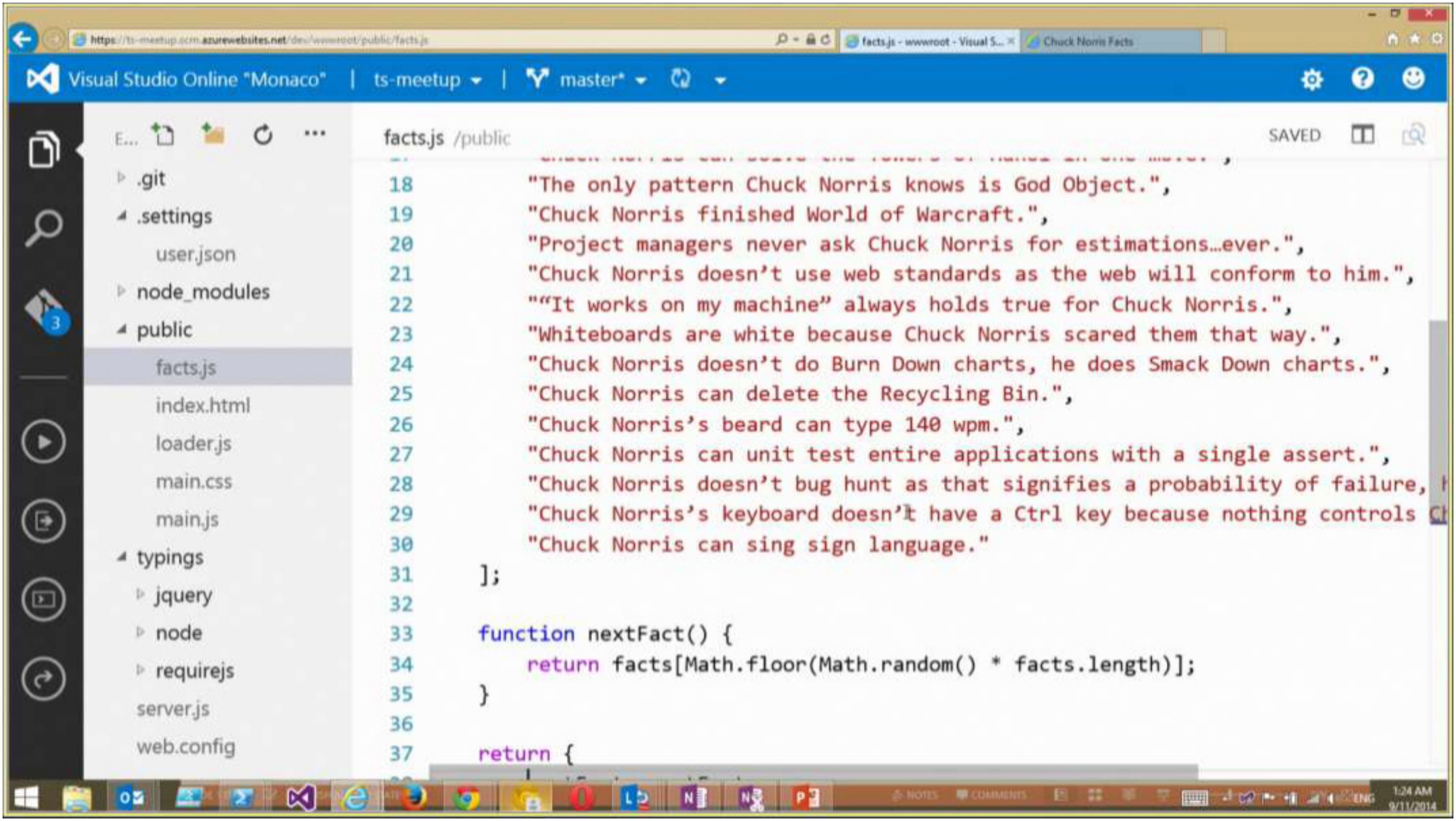
```
4 var http = require('http');
5 var url = require('url');
6 var path = require('path');
7
8 // create server
9 http.createServer(requestHandler).listen(
10
11 // handle requests
12 function requestHandler(req, res) {
13
14     var filepath = url.parse(req.url).pathname;
15     var fullpath = path.resolve(__dirname, filepath);
16
17     if (filepath === '/') {
18         filepath = 'index.html';
19     }
20
21     fullpath = path.join(__dirname, 'public', filepath);
22     http.stream(fullpath, res);
23 }
```

[info] The logging level is 'Information'. To change this, configure your application from <https://manage.windowsazure.com/#Workspaces/WebsiteExtension/Website/ts-meetup/configure>

[info] Restarting server...

[info] Click on <http://ts-meetup.azurewebsites.net> to open your site





Visual Studio Online "Monaco"

ts-meetup

master



E... ...

▶ .git

▲ .settings

user.json

▶ node_modules

▲ public

facts.js

index.html

loader.js

main.css

main.js

▲ typings

▶ jquery

▶ node

▶ requirejs

server.js

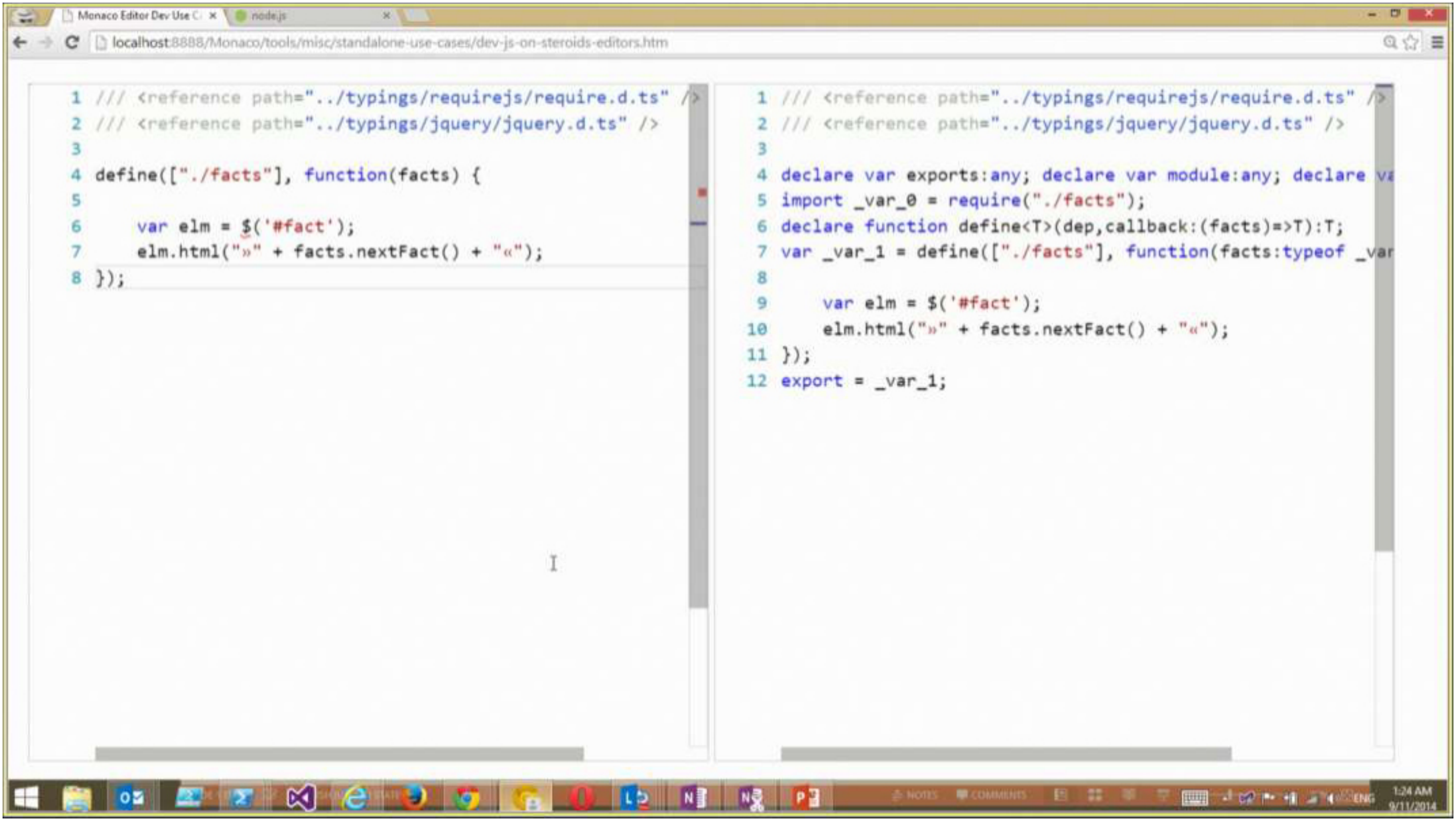
web.config

facts.js /public

SAVED

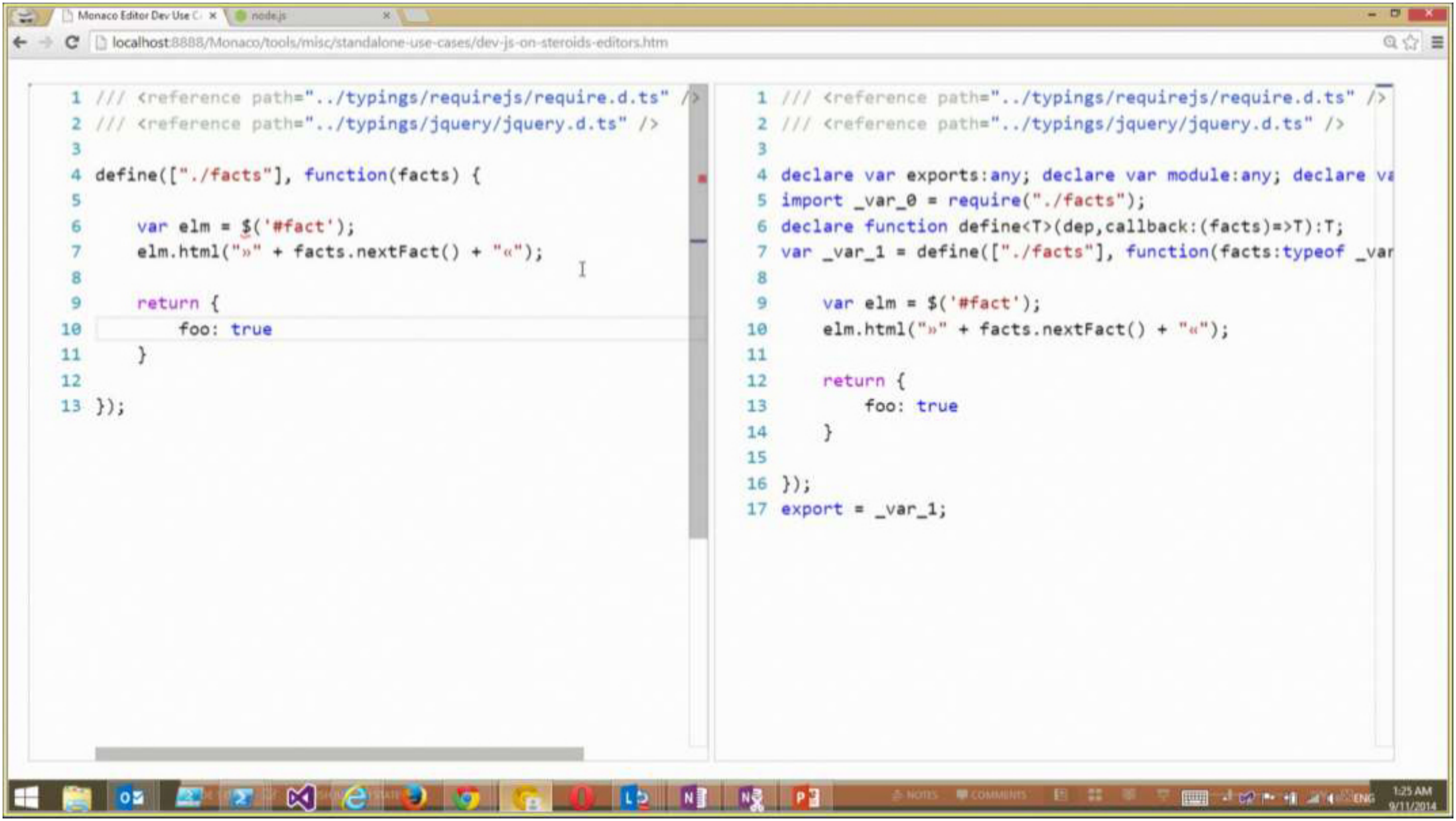


```
18 "The only pattern Chuck Norris knows is God Object.",
19 "Chuck Norris finished World of Warcraft.",
20 "Project managers never ask Chuck Norris for estimations...ever.",
21 "Chuck Norris doesn't use web standards as the web will conform to him.",
22 "'It works on my machine' always holds true for Chuck Norris.",
23 "Whiteboards are white because Chuck Norris scared them that way.",
24 "Chuck Norris doesn't do Burn Down charts, he does Smack Down charts.",
25 "Chuck Norris can delete the Recycling Bin.",
26 "Chuck Norris's beard can type 140 wpm.",
27 "Chuck Norris can unit test entire applications with a single assert.",
28 "Chuck Norris doesn't bug hunt as that signifies a probability of failure,
29 "Chuck Norris's keyboard doesn't have a Ctrl key because nothing controls
30 "Chuck Norris can sing sign language."
31 ];
32
33 function nextFact() {
34     return facts[Math.floor(Math.random() * facts.length)];
35 }
36
37 return {
```



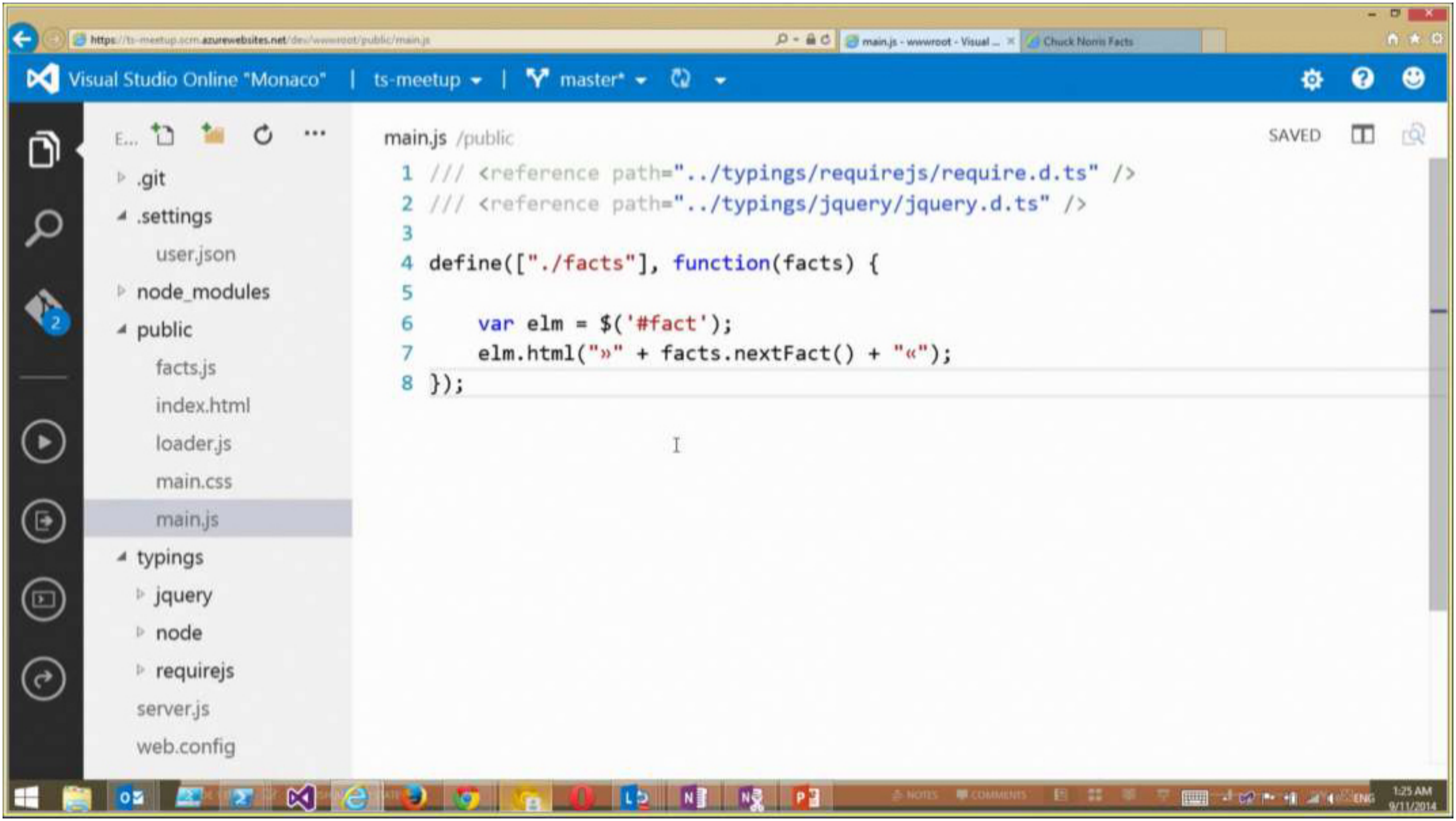
```
1 /// <reference path="../../typings/requirejs/require.d.ts" />
2 /// <reference path="../../typings/jquery/jquery.d.ts" />
3
4 define(["./facts"], function(facts) {
5
6     var elm = $('#fact');
7     elm.html(">" + facts.nextFact() + "<");
8 });
```

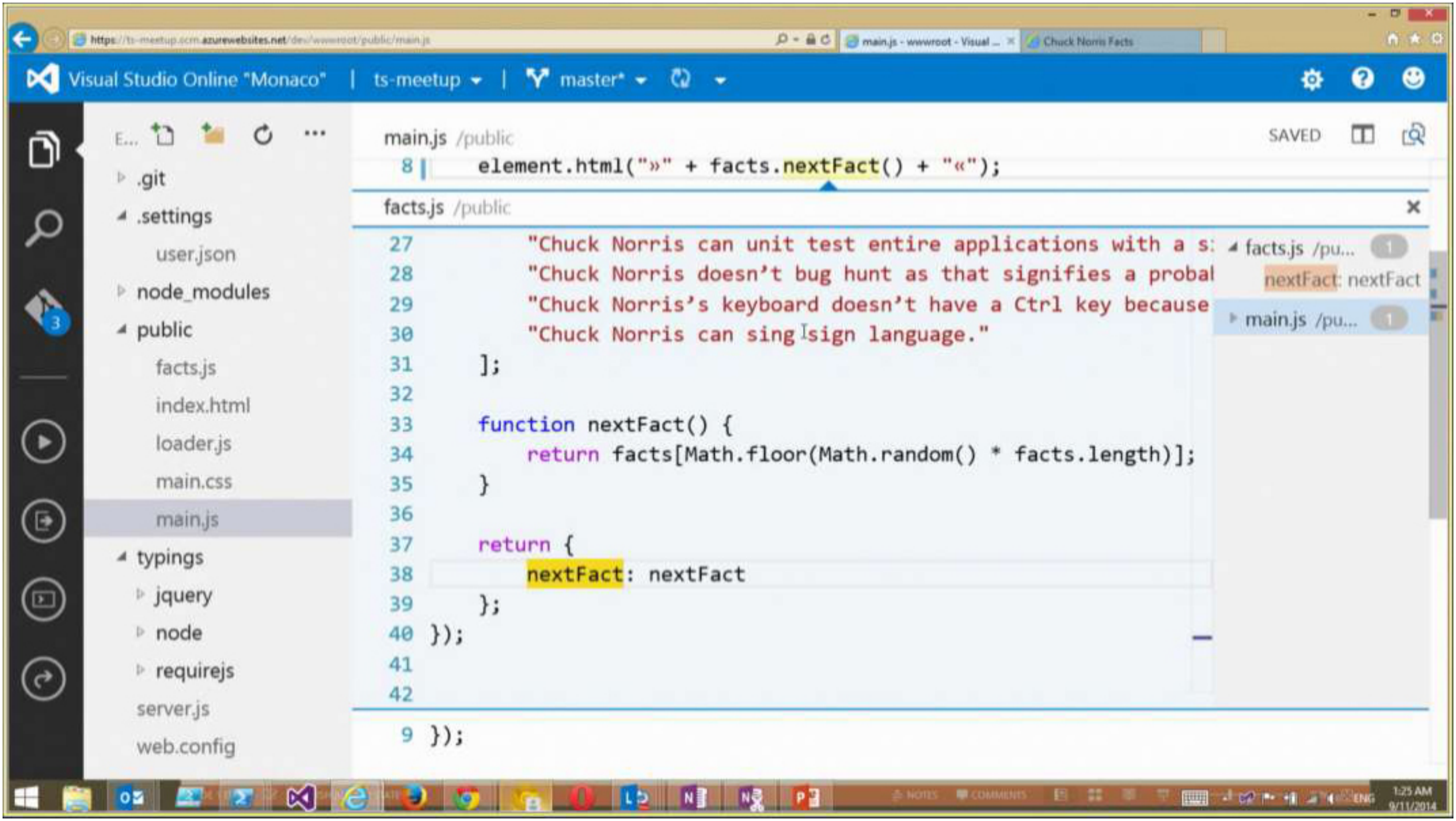
```
1 /// <reference path="../../typings/requirejs/require.d.ts" />
2 /// <reference path="../../typings/jquery/jquery.d.ts" />
3
4 declare var exports:any; declare var module:any; declare var __esModule:boolean;
5 import _var_0 = require("./facts");
6 declare function define<T>(dep,callback:(facts)=>T):T;
7 var _var_1 = define(["./facts"], function(facts:typeof _var_0) {
8
9     var elm = $('#fact');
10    elm.html(">" + facts.nextFact() + "<");
11 });
12 export = _var_1;
```

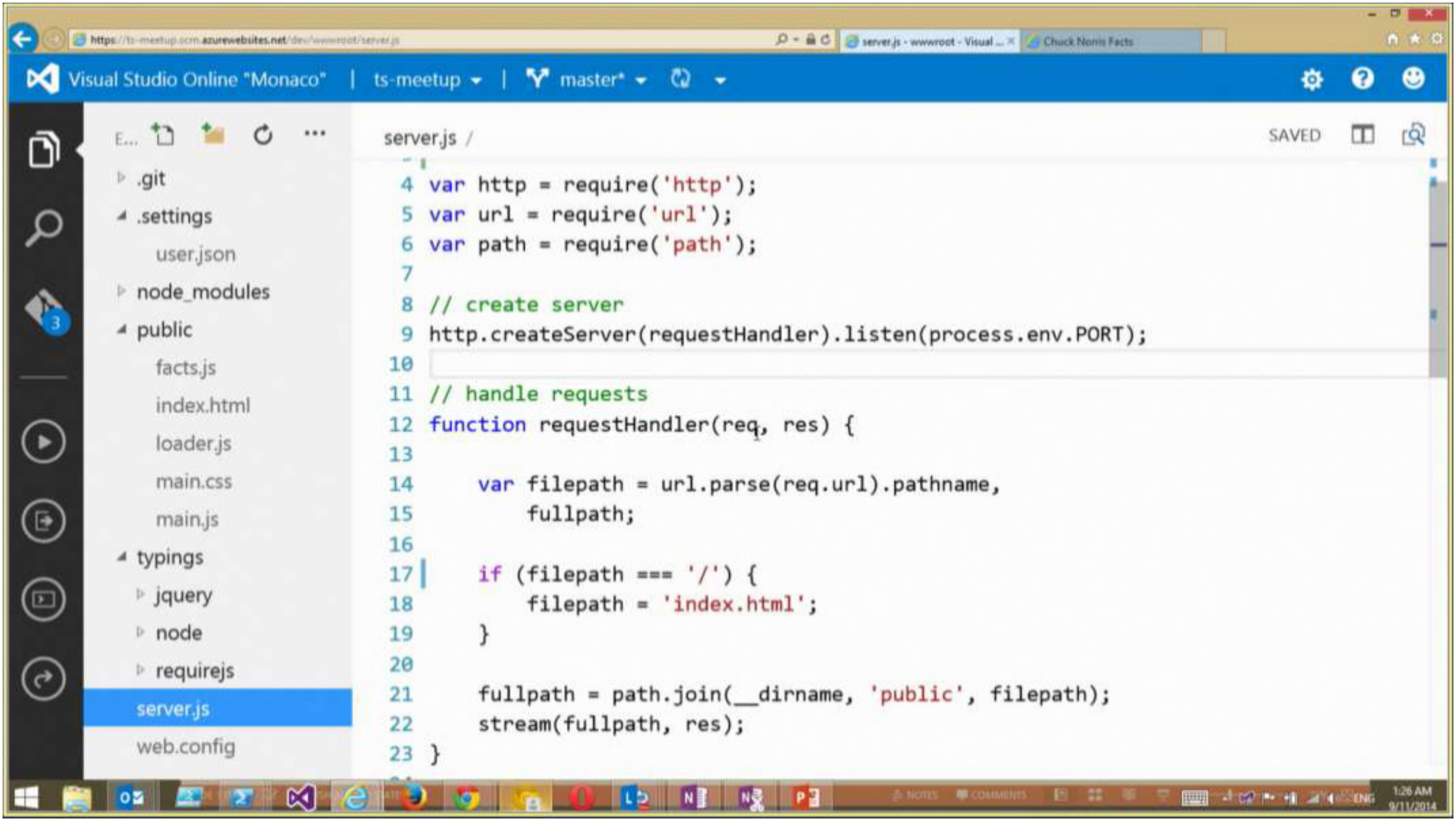



```
1 /// <reference path="../../typings/requirejs/require.d.ts" />
2 /// <reference path="../../typings/jquery/jquery.d.ts" />
3
4 define(["./facts"], function(facts) {
5
6     var elm = $('#fact');
7     elm.html(">" + facts.nextFact() + "<");
8
9     return {
10         foo: true
11     }
12
13 });
```

```
1 /// <reference path="../../typings/requirejs/require.d.ts" />
2 /// <reference path="../../typings/jquery/jquery.d.ts" />
3
4 declare var exports:any; declare var module:any; declare var
5 import _var_0 = require("./facts");
6 declare function define<T>(dep,callback:(facts)=>T):T;
7 var _var_1 = define(["./facts"], function(facts:typeof _var_0) {
8
9     var elm = $('#fact');
10    elm.html(">" + facts.nextFact() + "<");
11
12    return {
13        foo: true
14    }
15
16 });
17 export = _var_1;
```





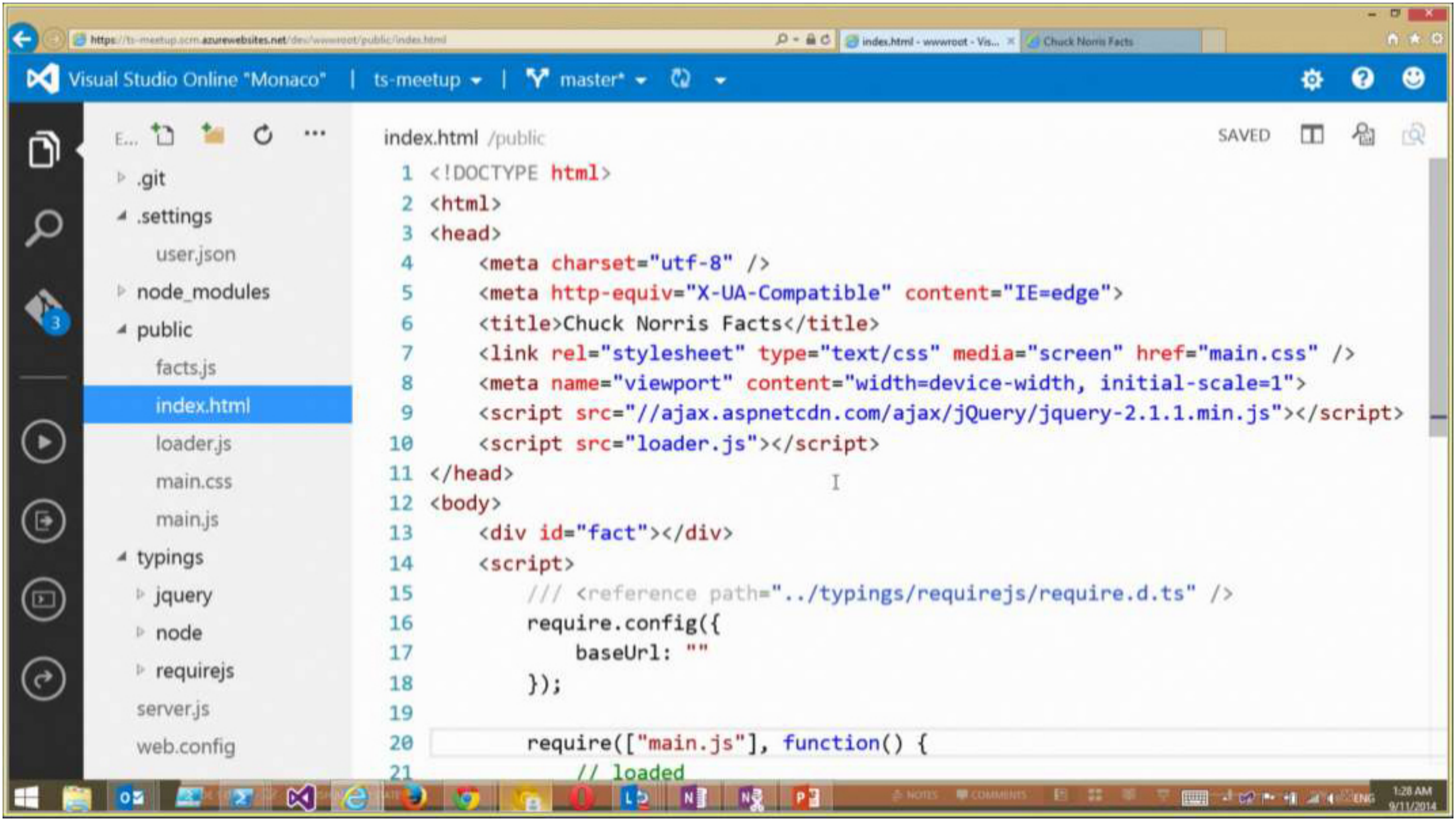


- E...
- .git
- .settings
 - user.json
- node_modules
- public
 - facts.js
 - index.html
 - loader.js
 - main.css
 - main.js
- typings
 - jquery
 - node
 - requirejs
- server.js
- web.config

server.js /

```
4 var http = require('http');
5 var url = require('url');
6 var path = require('path');
7
8 // create server
9 http.createServer(requestHandler).listen(process.env.PORT);
10
11 // handle requests
12 function requestHandler(req, res) {
13
14     var filepath = url.parse(req.url).pathname,
15         fullpath;
16
17     if (filepath === '/') {
18         filepath = 'index.html';
19     }
20
21     fullpath = path.join(__dirname, 'public', filepath);
22     stream(fullpath, res);
23 }
```

SAVED



Skype ECS UI Framework

Alexander Ryzhov

Skype ECS UI Development Approach



HTML Templates +
attached JavaScript

- Messy HTML and JavaScript code
- Hinders reusability, promotes **copy/paste**
- Used in web sites because of SEO and legacy browsers



Pure JavaScript app

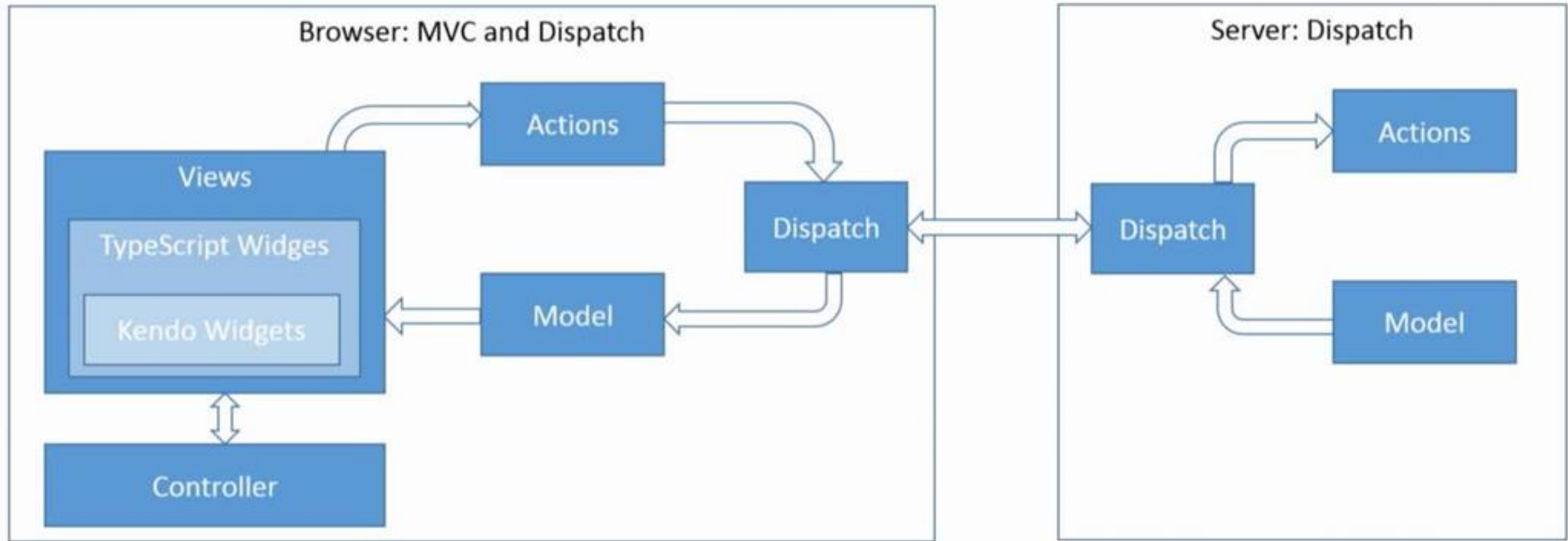


- Unlimited UI complexity
- Clean, modular code
- Requires better coding skills

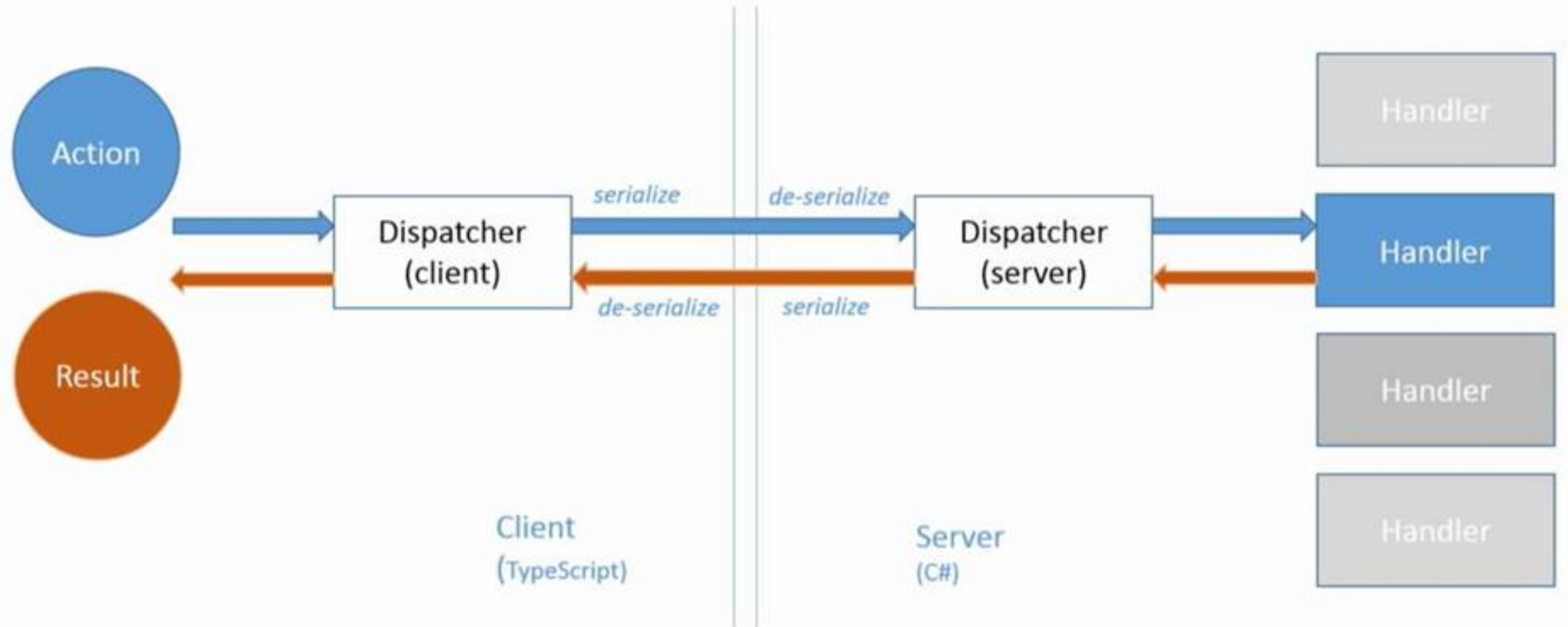
ECS UI Framework

- Extensible widget library
- MVC with nested controllers and history management using #
- AJAX-based Action/Dispatcher with centralized error handling
- Forms, data binding
- CSS-based styling

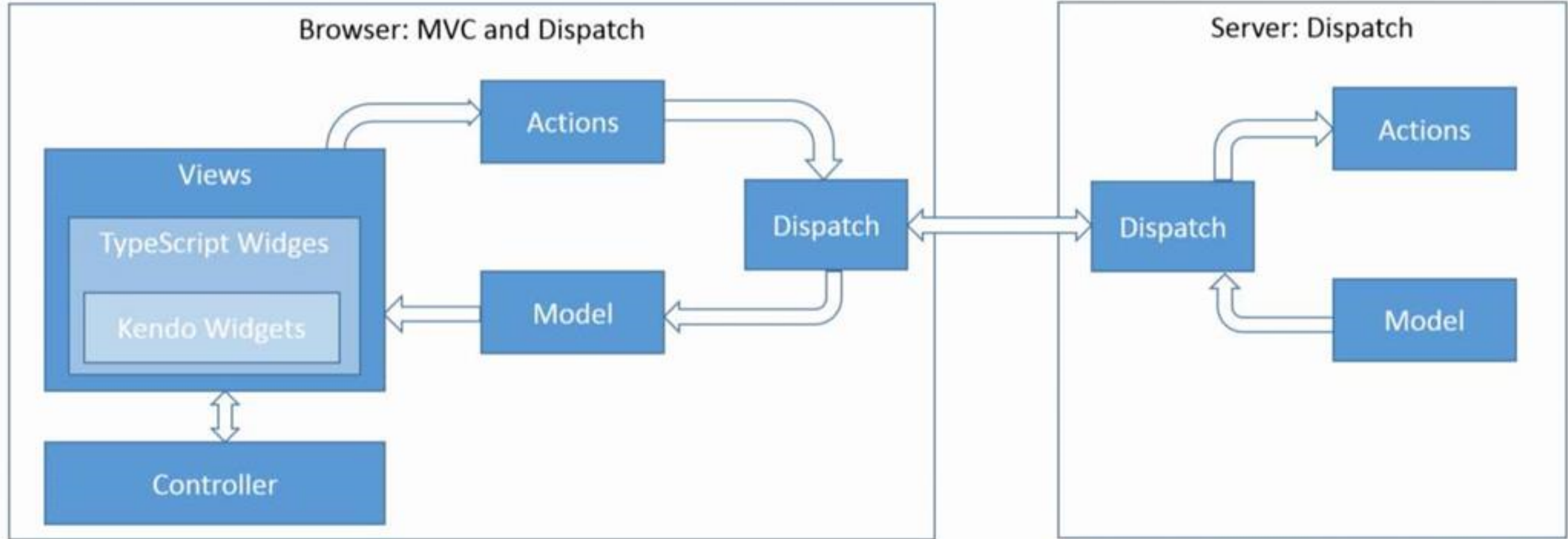
Skype ECS UI Architecture



Dispatcher



Skype ECS UI Architecture



ECS UI Framework

- Extensible widget library
- MVC with nested controllers and history management using #
- AJAX-based Action/Dispatcher with centralized error handling
- Forms, data binding
- CSS-based styling

Demo

jLib Framework

- Silverlight-like programming experience
- Xaml
- Simple data binding
- MVVM pattern
- Reusable control library
- Service Reference for oData
- Linq-to-oData from UI code

MSAsset

Search Items by Name

REDMONDj1length

Query

Locate

Add Item

Asset Lifecycle

Misc.

Bulk

SAVED SEARCHES (3)

msg X

157.55.150.62 X

by2msgweba01 X

blu002-tp137e03 X

server: chlazcmcu123... X

RECENT SEARCHES

Choose a DataSource: Asset - PhysicalAssetList

Filter Options

Index	Open	Column Name	Operator	Value	Close	Boolean
1						And
2						And

Add Filter X

Add Filter X

Select Columns for: Asset - PhysicalAssetList

Suppressions / Logical Associations

Below is a top 10 data preview.
Click Run Query to view full results.

Settings

Column Options

Refresh Preview

Page Size 25

Run Query

☐ Check All

☐ Show Checked

Default Columns

<input checked="" type="checkbox"/> AssetTag	<input type="checkbox"/> DataCenterLocation	<input type="checkbox"/> MasterId	<input type="checkbox"/> OSRevision
<input type="checkbox"/> AssetUnit	<input type="checkbox"/> DataCenterMasterId	<input type="checkbox"/> MasterSource	<input type="checkbox"/> OSServicePackVersion
<input type="checkbox"/> AuditModified	<input type="checkbox"/> DeploymentDate	<input type="checkbox"/> MaxWarrantyDate	<input type="checkbox"/> OSSystemDirectory
<input type="checkbox"/> AuditState	<input type="checkbox"/> DeviceState	<input checked="" type="checkbox"/> Model	<input type="checkbox"/> OSSystemDrive
<input type="checkbox"/> BinNumber	<input type="checkbox"/> DiscoveryAssetTag	<input type="checkbox"/> ModelId	<input type="checkbox"/> OSVersion
<input type="checkbox"/> BihApplicationId	<input type="checkbox"/> DiscoverySerialNumber	<input type="checkbox"/> Modified	<input checked="" type="checkbox"/> PropertyDimension
<input type="checkbox"/> BihPropertyGroupId	<input type="checkbox"/> DiscreteSKUName	<input type="checkbox"/> ModifiedBy	<input checked="" type="checkbox"/> PropertyDimensionId
<input type="checkbox"/> BusinessDivisionId	<input type="checkbox"/> DiscreteSKUNotes	<input checked="" type="checkbox"/> Name	<input type="checkbox"/> PropertyGroup
<input type="checkbox"/> BusinessDivisionName	<input type="checkbox"/> DiscreteSKUHeight	<input type="checkbox"/> NetworkDeviceType	<input type="checkbox"/> PropertyGroupId
<input type="checkbox"/> Chassis	<input type="checkbox"/> Environment	<input type="checkbox"/> NetworkDeviceTypeId	<input type="checkbox"/> Rack
<input type="checkbox"/> ChassisId	<input type="checkbox"/> EnvironmentId	<input type="checkbox"/> NetworkDomain	<input type="checkbox"/> RackId
<input type="checkbox"/> Colocation	<input type="checkbox"/> FinancialItemNumber	<input type="checkbox"/> OnlineBusinessId	<input type="checkbox"/> RackMasterId
<input type="checkbox"/> ColocationId	<input type="checkbox"/> FinancialMasterId	<input type="checkbox"/> OnlineBusinessName	<input type="checkbox"/> SecurityClassification
<input type="checkbox"/> ColocationMasterId	<input type="checkbox"/> iAdminPropertyDimensionId	<input type="checkbox"/> OnlineServiceId	<input checked="" type="checkbox"/> SerialNumber
<input type="checkbox"/> ColocationZone	<input type="checkbox"/> iAdminPropertyGroupId	<input type="checkbox"/> OnlineServiceName	<input type="checkbox"/> Side
<input type="checkbox"/> ColocationZoneId	<input checked="" type="checkbox"/> Id	<input type="checkbox"/> OperationalState	<input type="checkbox"/> SKUFinancialMasterId
<input type="checkbox"/> ColocationZoneMasterId	<input type="checkbox"/> IOld	<input type="checkbox"/> OSBuildNumber	<input type="checkbox"/> SKUName
<input checked="" type="checkbox"/> ConfigItemGroupDesc	<input type="checkbox"/> IsAssetOnly	<input type="checkbox"/> OSCountryCode	<input type="checkbox"/> SKUNotes
<input type="checkbox"/> Container	<input checked="" type="checkbox"/> IsMissionCritical	<input type="checkbox"/> OSLanguage	<input type="checkbox"/> SKUHeight
<input type="checkbox"/> ContainerId	<input type="checkbox"/> IsVirtual	<input type="checkbox"/> OSLocale	<input type="checkbox"/> SlotNumber
<input type="checkbox"/> ContainerMasterId	<input checked="" type="checkbox"/> ItemType	<input type="checkbox"/> OSMajorVersion	<input type="checkbox"/> SuppressionEndTime
<input type="checkbox"/> ContainsPII	<input type="checkbox"/> Location	<input type="checkbox"/> OSManufacturerId	<input type="checkbox"/> SuppressionReason
<input type="checkbox"/> Created	<input type="checkbox"/> LocationDesc	<input type="checkbox"/> OSMinorVersion	<input type="checkbox"/> SuppressionStartTime
<input type="checkbox"/> CreatedBy	<input type="checkbox"/> LocationId	<input type="checkbox"/> OSModelId	<input type="checkbox"/> SuppressionStatusCode
<input checked="" type="checkbox"/> DataCenter	<input checked="" type="checkbox"/> Manufacturer	<input type="checkbox"/> OSOrganizationalUnit	<input type="checkbox"/> SuppressionStatusDesc
<input checked="" type="checkbox"/> DataCenterCode	<input type="checkbox"/> ManufacturerId	<input type="checkbox"/> OSProductType	<input type="checkbox"/> Title
<input checked="" type="checkbox"/> DataCenterId			

AssetTag Model PropertyDimension PropertyDimensionId Name SerialNumber Id ConfigItemGroupDesc IsMissionCritical ItemType DataCenter Manufacturer DataCenterCode DataCenterId

Windows Taskbar

System Tray

Demo

- Demo1: MSAsset – MVVM
- Demo2: Hello World!
- Demo3: Simple Binding
- Demo4: Data Binding
- Demo5: Binding to collection
- Demo6: DataGrid hierarchy
- Demo7: <http://jlib>
- Demo8: oDataGrid

Data binding

WPF/Silverlight

```
public class Person : INotifyPropertyChanged
{
    private string name;
    // Declare the event
    public event PropertyChangedEventHandler PropertyChanged;

    public Person()
    {
    }

    public Person(string value)
    {
        this.name = value;
    }

    public string PersonName
    {
        get { return name; }
        set
        {
            name = value;
            // Call OnPropertyChanged whenever the property is updated
            OnPropertyChanged("PersonName");
        }
    }

    // Create the OnPropertyChanged method to raise the event
    protected void OnPropertyChanged(string name)
    {
        PropertyChangedEventHandler handler = PropertyChanged;
        if (handler != null)
        {
            handler(this, new PropertyChangedEventArgs(name));
        }
    }
}
```

TS/jLib

```
export class Person {
    public Name: jLib.BindableProperty<string> = new jLib.BindableProperty<string>();
    constructor() {}
}
```

oData Custom Action

.Net/oData Proxy

```
public List<string> GetAvailableIPAddresses(  
    string iprangeId,  
    string startAddress,  
    string endAddress,  
    int pageCount)  
{  
    OperationParameter[] operationParameter =  
        new OperationParameter[]  
        {  
            new BodyOperationParameter("rangeId", iprangeId),  
            new BodyOperationParameter("startSearchAddress", startAddress),  
            new BodyOperationParameter("endSearchAddress", endAddress),  
            new BodyOperationParameter("count", pageCount)  
        };  
  
    List<string> result = this.mfxGlobalAssetModel.Execute<string>(  
        new Uri(this.globalAssetSvcBaseUrl.AbsoluteUri + "/GetIPsFromIPRange"),  
        "POST",  
        false,  
        operationParameter).ToList();  
  
    result.Sort(CompareIPAddresses);  
  
    return result;  
}
```

TS/jLib

```
public static GetIPsFromIPRangeAsync(rangeId: number, startSearchAddress: string, endSearchAddress: string, count: number)
```


Markup, DataTemplate

```
<DataGrid x:Name="dataGrid" ItemsSource="{Binding HostedItems}" AllowPagination="true" PageSize="50" AllowSearchGrid="false">
  <DataGrid.ColumnTemplates>
    <!-- Name -->
    <DataGridTextColumn HeaderText="Name" Binding="{Binding Name}"/>
    <!-- ID -->
    <DataGridTemplateColumn HeaderText="ID">
      <DataGridTemplateColumn.CellTemplate>
        <DataTemplate>
          <Hyperlink Css.Name="detailsLink"
                    Content="{Binding Id}"
                    NavigateUri="#" />
        </DataTemplate>
      </DataGridTemplateColumn.CellTemplate>
    </DataGridTemplateColumn>
    <!-- Type -->
    <DataGridTextColumn HeaderText="Type" Binding="{Binding Type}"/>
  </DataGrid.ColumnTemplates>
</DataGrid>
```

- Download <http://codebox/jlib>
- Demo <http://jlib/>
- Contact jlib@Microsoft.com
- Hackathon Video
<https://onehack.azurewebsites.net/project/897220bd-650c-e411-903f-00155d5066d7>
- Whitepaper <\\msgscmds\STORAGE\Users\zifengh\jLibFramework\jLibFramework.docx>